More on Interpolation: Splines and Radial Basis Functions

M.Sc. in Mathematical Modelling & Scientific Computing,
Practical Numerical Analysis

Michaelmas Term 2019, Lecture 2

1D Interpolation

Recall the canonical 1D interpolation problem: given a set of nodes x_i, $0 \leq i \leq n$ and data at those nodes $f(x_i)$, construct a function $p(x)$ such that

$$p(x_i) \quad = \quad f(x_i)$$

for $0 \leq i \leq n$.

Last time we looked at the Lagrange interpolant which was global, in the sense that it was defined by the same function on the whole interval.

This time we look for piecewise polynomial interpolants — functions which are polynomials on each subinterval $[x_{i-1}, x_i]$, $1 \leq i \leq n$, and satisfy certain continuity conditions. These are known as splines.

Splines

Again we are given a set of nodes x_i, $0 \leq i \leq n$ and data at those nodes $f(x_i)$. When talking about splines, the x_i are often known as knots. The interpolating spline:

- is a polynomial of degree k in each subinterval $[x_{i-1}, x_i]$, $1 \leq i \leq n$;
- is continuous and has continuous derivatives up to order $k-1$;
- satisfies the interpolation conditions.

Linear Splines

The simplest splines are linear splines (i.e. $k = 1$). The continuity and interpolation conditions are enough to determine them uniquely since

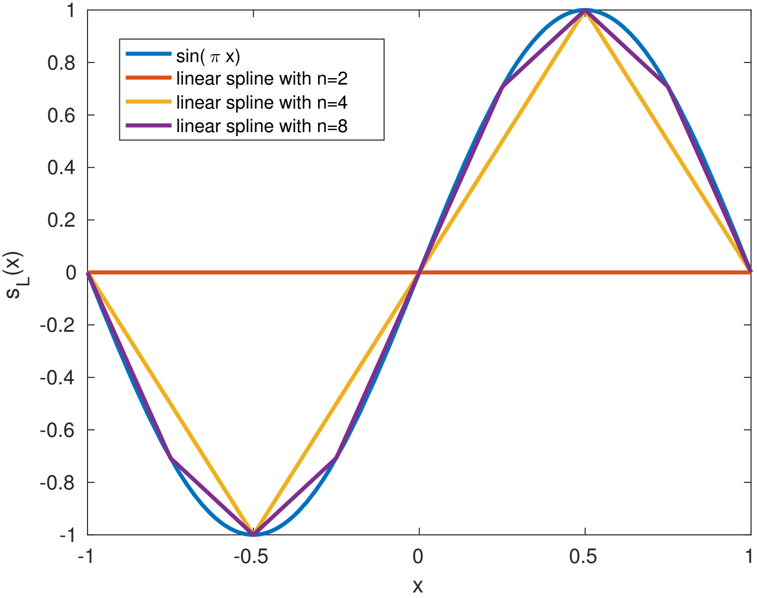
- the number of unknowns is $2n$ (there are n intervals and 2 unknowns required to determine a linear function in each subinterval)
- the number of constraints is $2n$ made up of
  - $n + 1$ interpolation conditions (at x_i, $0 \leq i \leq n$)
  - $n - 1$ continuity conditions (at x_i, $1 \leq i \leq n - 1$)

We can write the linear spline, $s_L(x)$, as

$$s_L(x) \quad = \quad \frac{x_i - x}{x_i - x_{i-1}} f(x_{i-1}) + \frac{x - x_{i-1}}{x_i - x_{i-1}} f(x_i)$$

for $x \in [x_{i-1}, x_i]$, $1 \leq i \leq n$.

Example



sin(π x)
linear spline with n=2
linear spline with n=4
linear spline with n=8
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
s_L(x)
-1
-0.5
0
0.5
1
x

Convergence

Theorem 1

Suppose $f \in C^2[a,b]$ and let $s_L(x)$ be the linear spline that interpolates f at the knots $a = x_0 < x_1 < \ldots < x_n = b$, then

$$\|f - s_L\|_\infty \quad \leq \quad \frac{1}{8} h^2 \|f''\|_\infty \,.$$

where $h = \max_i h_i$ and $h_i = x_i - x_{i-1}$.

This tells us that if we use a uniform grid, then every time we double n (and thus we halve h) we expect the error to decrease by a factor of 4.

Minimisation Property

The linear spline also has a nice minimisation property as follows:

Theorem 2

Let s_L be the linear spline that interpolates a function $f \in C[a,b]$ at the knots $a = x_0 < x_1 < \ldots < x_n = b$. Then for any function v in $H^1(a,b)$ that also interpolates f at the knots,

$$\|s_L'\|_2 \;\leq\; \|v'\|_2 \,.$$

In other words, this theorem tells us that, among all functions in $H^1(a,b)$ that interpolate f at the knots, the linear spline $s_L(x)$ is the flattest, in the sense that its average slope is the smallest.

Global Form of Linear Splines

We may also write a global expression for the linear interpolating spline as a sum of basis functions:

$$s_L(x) = \sum_{i=0}^{n} \phi_i(x) f(x_i) \,,$$

where the basis functions $\phi_i(x)$ are defined as

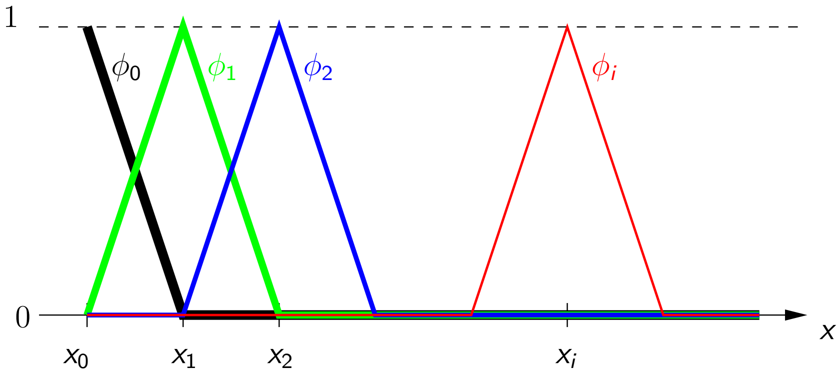
$$\phi_i(x) = \begin{cases} \frac{x - x_{i-1}}{x_i - x_{i-1}} & \text{if } x_{i-1} \le x \le x_i \\ \frac{x_{i+1} - x}{x_{i+1} - x_i} & \text{if } x_i \le x \le x_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

for $1 \le i \le n$ and

$$\phi_0(x) = \begin{cases} \frac{x_1 - x}{x_1 - x_0} & \text{if } x_0 \le x \le x_1 \\ 0 & \text{otherwise} \end{cases}$$

$$\phi_n(x) = \begin{cases} \frac{x - x_{n-1}}{x_n - x_{n-1}} & \text{if } x_{n-1} \le x \le x_n \\ 0 & \text{otherwise} \end{cases}$$

Basis Functions



Note that the basis functions are nodal, i.e. $\phi_i(x_j) = \delta_{i,j}$. Thus

$$s_L(x_j) \;=\; \sum_{i=0}^{n} \phi_i(x_j) f(x_i) \;=\; \sum_{i=0}^{n} \delta_{i,j} f(x_i) \;=\; f(x_j)\;.$$

Cubic Splines

Cubic splines are also popular due to their increased regularity over linear splines. This time

- the number of unknowns is $4n$ (there are n intervals and 4 unknowns required to determine a cubic function in each subinterval)
- the number of constraints is $4n - 2$ made up of
  - $n + 1$ interpolation conditions (at x_i, $0 \leq i \leq n$)
  - $3(n - 1)$ continuity conditions ($s(x)$, $s'(x)$ and $s''(x)$ must be continuous at x_i, $1 \leq i \leq n - 1$)

Thus we need two more conditions to determine the cubic spline uniquely.

Natural Cubic Splines

For natural cubic splines the two final conditions are

$$s''(x_0) \;=\; s''(x_n) \;=\; 0\;.$$

Such splines have a minimisation property analagous to linear splines and are characterised as follows:

Theorem 3

Let s be the natural cubic spline that interpolates a function $f \in C[a,b]$ at the knots $a = x_0 < x_1 < \ldots < x_n = b$. Then for any function v in $H^2(a,b)$ that also interpolates f at the knots,

$$\|s''\|_2 \;\leq\; \|v''\|_2\;.$$

This theorem essentially means that the natural cubic spline minimises the 'average curvature' over functions in $H^2(a,b)$ that interpolate f at the knots.

Construction of Natural Cubic Splines

Let $\sigma_i = s''(x_i)$ for $0 \le i \le n$ (note these are unknown). Then we can write

$$s''(x) \;=\; \frac{x_i - x}{h_i}\sigma_{i-1} + \frac{x - x_{i-1}}{h_i}\sigma_i\,, \quad \text{for } x \in [x_{i-1}, x_i]\,.$$

Integrate twice to get

$$s(x) \;=\; \frac{(x_i - x)^3}{6h_i}\sigma_{i-1} + \frac{(x - x_{i-1})^3}{6h_i}\sigma_i + \alpha_i(x - x_{i-1}) + \beta_i(x_i - x)\,,$$

for $x \in [x_{i-1}, x_i]$. Here the α_i and β_i are constants of integration to be determined. The interpolation conditions become

$$s(x_{i-1}) \;=\; \frac{1}{6}\sigma_{i-1}h_i^2 + h_i\beta_i \;=\; f(x_{i-1})\,, \qquad (1)$$

$$s(x_i) \;=\; \frac{1}{6}\sigma_i h_i^2 + h_i\alpha_i \;=\; f(x_i)\,. \qquad (2)$$

Construction of Natural Cubic Splines

Using the interpolation conditions, the definition of $s(x)$ and the continuity of s' at the knots gives, after some algebra,

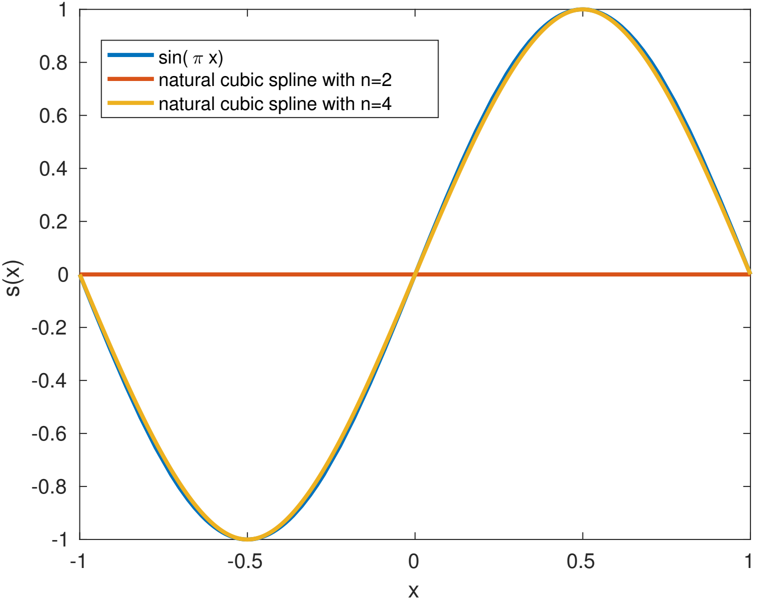
$$h_i\sigma_{i-1} + 2(h_{i+1} + h_i)\sigma_i + h_{i+1}\sigma_{i+1} = 6\left(\frac{f(x_{i+1}) - f(x_i)}{h_{i+1}} - \frac{f(x_i) - f(x_{i-1})}{h_i}\right)$$

along with

$$\sigma_0 = \sigma_n = 0$$

which is a nonsingular tridiagonal system for the σ_i. Once we know the σ_i we can also compute the α_i and the β_i coefficients, using (1) and (2), and hence the natural cubic spline in each subinterval.

Example



sin(π x)
natural cubic spline with n=2
natural cubic spline with n=4
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
s(x)
-1
-0.5
0
0.5
1
x

Higher Dimensions

Lots of different options:

- Tensor product grids (see the Chebfun project)
- Piecewise polynomials on polygons (often triangles or rectangles in 2D) — extension to differential equations yields finite element method
- Radial basis functions

We'll take a look at radial basis functions.

Radial Basis Function Interpolation

Again we wish to solve the canonical interpolation problem: given a set of nodes $\mathbf{x}_i$, $0 \leq i \leq n$ and data at those nodes $f(\mathbf{x}_i)$, construct a function $p(\mathbf{x})$ such that

$$p(\mathbf{x}_i) \quad = \quad f(\mathbf{x}_i)$$

for $0 \leq i \leq n$.

We write

$$p(\mathbf{x}) \quad = \quad \sum_{j=0}^{n} \alpha_j \phi\left(\frac{\|\mathbf{x}-\mathbf{x}_j\|_2}{\delta}\right) \; .$$

In other words, we write $p(x)$ as a sum of basis functions as we did in the 1D polynomial interpolation case but this time the basis functions depend on the distance to each node and a parameter δ. We also need to define the function $\phi : [0, \infty) \to \mathbb{R}$.

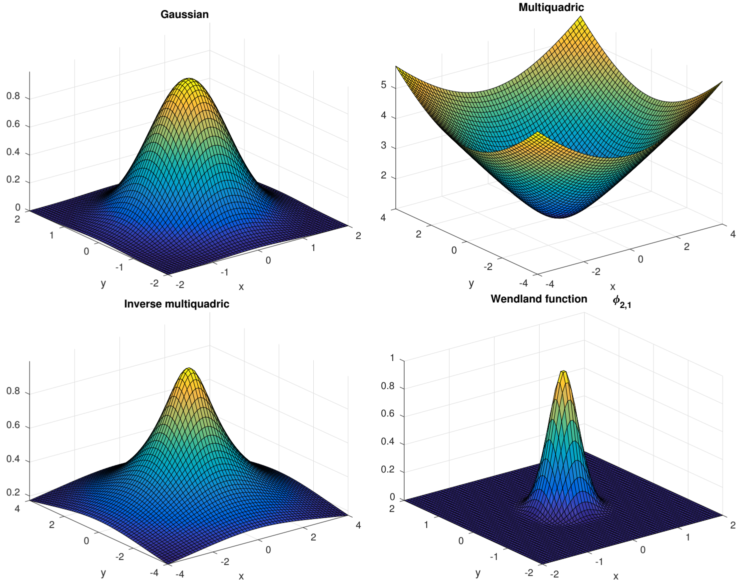
Example Basis Functions

Generally the functions ϕ are chosen to have a maximum or minimum at $r = 0$ and to be of one sign.

Common basis functions include:

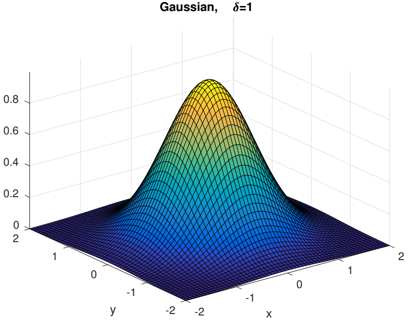
- Gaussians: $\phi(r) = \mathrm{e}^{-r^2}$
- Multiquadrics: $\phi(r) = \sqrt{1 + r^2}$
- Inverse multiquadrics: $\phi(r) = 1/\sqrt{1 + r^2}$
- Compactly supported functions (Wendland functions)

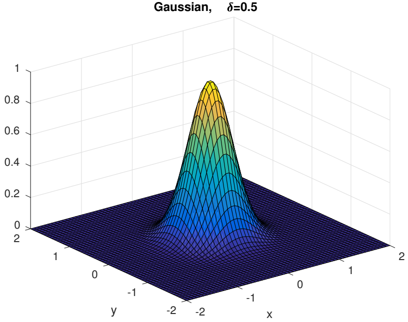
Example Basis Functions



Role of δ

The role of δ is essentially to adjust how far the basis function spreads — larger δ means the function spreads out more.





Compactly Supported RBFs

The compactly supported functions were constructed to be polynomials on the interval $[0, 1]$ and zero outside that interval. The polynomial part has minimal degree so that the global function has a given smoothness (i.e. so that the derivatives are continuous at $r = 1$). In two space dimensions we have

Function	Smoothness
$\phi_{2,1} = (1-r)_+^4(4r+1)$	C^2
$\phi_{2,2} = (1-r)_+^6(35r^2+18r+3)$	C^4
$\phi_{2,3} = (1-r)_+^8(32r^3+25r^2+8r+1)$	C^6

Here we define the notation x_+ by $x_+ = \max(x, 0)$.

Interpolation

The interpolation problem now works as in 1D. We have

$$p(\mathbf{x}_i) \;=\; \sum_{j=0}^{n} \alpha_j \phi\left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2}{\delta}\right) \;=\; f(\mathbf{x}_i)\;,$$

for $0 \leq i \leq n$. Thus we solve $A\boldsymbol{\alpha} = \mathbf{f}$ where the entries of the (symmetric) matrix A are given by

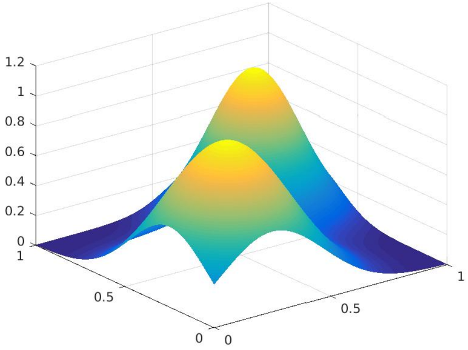
$$A_{i,j} \;=\; \phi\left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2}{\delta}\right)\;,$$

for $0 \leq i, j \leq n$.

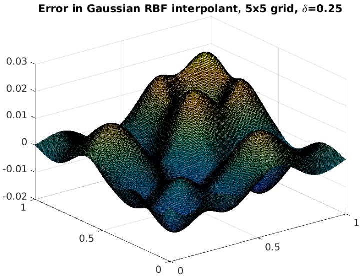
Example

We interpolate the function

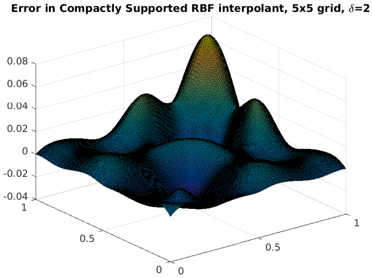
$$f(x,y) = \mathrm{e}^{-10((x-0.25)^2+(y-0.25)^2)} + \mathrm{e}^{-20((x-0.75)^2+(y-0.75)^2)} .$$



Example



Error in Gaussian RBF interpolant, 5x5 grid, δ=0.25
0.03
0.02
0.01
0
-0.01
-0.02
1
0.5
0
0
0.5
1



Error in Compactly Supported RBF interpolant, 5x5 grid, δ=2
0.08
0.06
0.04
0.02
0
-0.02
-0.04
1
0.5
0
0
0.5
1