Chapter 10

Function approximation

We have studied methods for computing solutions to algebraic equations in the form of real numbers or finite dimensional vectors of real numbers. In contrast, solutions to differential equations are scalar or vector valued functions, which only in simple special cases are analytical functions that can be expressed by a closed mathematical formula.

Instead we use the idea to approximate general functions by linear combinations of a finite set of simple analytical functions, for example trigonometric functions, splines or polynomials, for which attractive features are orthogonality and locality. We focus in particular on piecewise polynomials defined by the finite set of nodes of a mesh, which exhibit both near orthogonality and local support.

10.1 Function approximation

The Lebesgue space $L^2(I)$

Inner product spaces provide tools for approximation based on orthogonal projections on subspaces. We now introduce an inner product space for functions on the interval $I = [a, b]$, the *Lebesgue space* $L_2(I)$, defined as the class of all square integrable functions $f : I \to \mathbb{R}$,

$$L^2(I) = \{f : \int_a^b |f(x)|^2 \, dx < \infty\}. \tag{10.1}$$

The vector space $L^2(I)$ is closed under the basic operations of pointwise addition and scalar multiplication, by the inequality,

$$(a + b)^2 \leq 2(a^2 + b^2), \quad \forall a, b \geq 0, \tag{10.2}$$

which follows from *Young's inequality.*

Theorem 17 (Young's inequality)**.** *For $a, b \geq 0$ and $\epsilon > 0$,*

$$ab \leq \frac{1}{2\epsilon}a^2 + \frac{\epsilon}{2}b^2 \tag{10.3}$$

Proof. $0 \leq (a - \epsilon b)^2 = a^2 + \epsilon^2 b^2 - 2ab\epsilon$. □

The L^2-inner product is defined by

$$(f, g) = (f, g)_{L^2(I)} = \int_a^b f(x)g(x)\,dx, \tag{10.4}$$

with the associated L^2 norm,

$$\|f\| = \|f\|_{L^2(I)} = (f, f)^{1/2} = \left(\int_a^b |f(x)|^2\,dx\right)^{1/2}, \tag{10.5}$$

for which the Cauchy-Schwarz inequality is satisfied,

$$(f, g) \leq \|f\|\|g\|. \tag{10.6}$$

Approximation of functions in $L^2(I)$

We seek to approximate a function f in a vector space V by a linear combination of functions $\phi_j \in V$, that is

$$f(x) \approx f_n(x) = \sum_{j=1}^{n} \alpha_j \phi_j(x), \tag{10.7}$$

with $\alpha_j \in \mathbb{R}$. If linearly independent, the set $\{\phi_j\}_{j=1}^n$ spans a subspace $S \subset V$, that is

$$S = \{f_n \in V : \ f_n = \sum_{j=1}^{n} \alpha_j \phi_j(x),\ \alpha_j \in \mathbb{R}\}, \tag{10.8}$$

with the set $\{\phi_j\}_{j=1}^n$ a basis for S. For example, in a *Fourier series* the basis functions ϕ_j are trigonometric functions, in a *power series* monomials.

The question is now how to determine the coordinates α_j so that $f_n(x)$ is a good approximation of $f(x)$ in the subspace S. One approach to the problem is to use the techniques of orthogonal projections previously studied for vectors in $\mathbb{R}^n$, an alternative approach is interpolation, where α_j are chosen such that $f_n(x_i) = f(x_i)$, in a set of nodes x_i, for $i = 1, ..., n$. If we cannot evaluate the function $f(x)$ in arbitrary points x, but only have access to a set of sampled data points $\{(x_i, f_i)\}_{i=1}^m$, with $m > n$, we can formulate a least squares problem to determine the coordinates α_j that minimize the error $f(x_i) - f_i$, in a suitable norm.

L^2 projection

The L^2 *projection* Pf, onto the subspace $S \subset V$, defined by (10.8), of a function $f \in V$, with $V = L^2(I)$, is the orthogonal projection of f on S, that is,

$$(f - Pf, s) = 0, \quad \forall s \in S, \tag{10.9}$$

which corresponds to,

$$\sum_{j=1}^{n} \alpha_j (\phi_i, \phi_j) = (f, \phi_i), \quad \forall i = 1, ..., n. \tag{10.10}$$

By solving the matrix equation $Ax = b$, with $a_{ij} = (\phi_i, \phi_j)$, $x_j = \alpha_j$, and $b_i = (f, \phi_i)$, we obtain the L_2 projection as

$$Pf(x) = \sum_{j=1}^{n} \alpha_j \phi_j(x). \tag{10.11}$$

We note that if $\phi_i(x)$ has *local support*, that is $\phi_i(x) \neq 0$ only for a subinterval of I, then the matrix A is sparse, and for $\{\phi_i\}_{i=1}^n$ an orthonormal basis, A is the identity matrix with $\alpha_j = (f, \phi_j)$.

Interpolation

The *interpolant* $\pi f \in S$, is determined by the condition that $\pi f(x_i) = f(x_i)$, for n nodes $\{x_i\}_{i=1}^n$. That is,

$$f(x_i) = \pi f(x_i) = \sum_{j=1}^{n} \alpha_j \phi_j(x_i), \quad i = 1, ..., n, \tag{10.12}$$

which corresponds to matrix equation $Ax = b$, with $a_{ij} = \phi_j(x_i)$, $x_j = \alpha_j$, and $b_i = f(x_i)$.

The matrix A is an identity matrix under the condition that $\phi_j(x_i) = 1$, for $i = j$, and zero else. We then refer to $\{\phi_i\}_{i=1}^n$ as a *nodal basis*, for which $\alpha_j = f(x_j)$ and we can express the interpolant as

$$\pi f(x) = \sum_{j=1}^{n} \alpha_j \phi_j(x) = \sum_{j=1}^{n} f(x_j) \phi_j(x). \tag{10.13}$$

Regression

If we cannot evaluate the function $f(x)$ in arbitrary points, but only have access to a set of data points $\{(x_i, f_i)\}_{i=1}^m$, with $m > n$, we can formulate the least squares problem,

$$\min_{f_n \in S} \|f_i - f_n(x_i)\| = \min_{\{\alpha_j\}_{j=1}^n} \|f_i - \sum_{j=1}^n \alpha_j \phi_j(x_i)\|, \quad i = 1, ..., m, \tag{10.14}$$

which we can solve by constructing the normal equations.

10.2 Piecewise polynomial approximation

Polynomial spaces

We introduce the vector space $\mathcal{P}^q(I)$, defined by the set of polynomials

$$p(x) = \sum_{i=0}^q c_i x^i, \tag{10.15}$$

of at most order q on an interval $I \in \mathbb{R}$, with the basis functions x^i and coordinates c_i, and the basic operations of pointwise addition and scalar multiplication,

$$(p + r)(x) = p(x) + r(x), \quad (\alpha p)(x) = \alpha p(x), \tag{10.16}$$

for $p, r \in \mathcal{P}^q(I)$ and $\alpha \in \mathbb{R}$. One basis for $\mathcal{P}^q(I)$ is the set of *monomials* $\{x^i\}_{i=0}^q$, another is $\{(x - c)^i\}_{i=0}^q$ which gives the *power series*,

$$p(x) = \sum_{i=0}^q a_i (x - c)^i = a_0 + a_1(x - c) + ... + a_q(x - c)^q, \tag{10.17}$$

for $c \in I$, with a Taylor series being an example of a power series,

$$f(x) = f(y) + f'(y)(x - y) + \frac{1}{2} f''(y)(x - y)^2 + ... \tag{10.18}$$

Langrange polynomials

For a set of nodes $\{x_i\}_{i=0}^q$, we define the *Lagrange polynomials* $\{\lambda\}_{i=0}^q$, by

$$\lambda_i(x) = \frac{(x - x_0) \cdots (x - x_{i-1})(x - x_{i+1}) \cdots (x - x_q)}{(x_i - x_0) \cdots (x_i - x_{i-1})(x_i - x_{i+1}) \cdots (x_i - x_q)} = \prod_{i \neq j} \frac{x - x_j}{x_i - x_j},$$

that constitutes a basis for $\mathcal{P}^q(I)$, and we note that

$$\lambda_i(x_j) = \delta_{ij}, \tag{10.19}$$

with the *Dirac delta function* defined as

$$\delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \tag{10.20}$$

so that $\{\lambda\}_{i=0}^q$ is a nodal basis, which we refer to as the *Lagrange basis*. We can express any $p \in \mathcal{P}^q(I)$ as

$$p(x) = \sum_{i=1}^q p(x_i)\lambda_i(x), \tag{10.21}$$

and by (10.13) we can define the *polynomial interpolant* $\pi_q f \in \mathcal{P}^q(I)$,

$$\pi_q f(x) = \sum_{i=1}^q f(x_i)\lambda_i(x), \quad x \in I, \tag{10.22}$$

for a continuous function $f \in \mathcal{C}(I)$.

Piecewise polynomial spaces

We now introduce piecewise polynomials defined over a partition of the interval $I = [a, b]$,

$$a = x_0 < x_1 < \cdots < x_{m+1} = b, \tag{10.23}$$

for which we let the *mesh* $\mathcal{T}_h = I_i$ denote the set of subintervals $I_j = (x_{i-1}, x_i)$ of length $h_i = x_i - x_{i-1}$, with the *mesh function*,

$$h(x) = h_i, \quad \text{for } x \in I_i. \tag{10.24}$$

We now define two vector spaces of *piecewise polynomials*, the discontinuous piecewise polynomials on I, defined by

$$W_h^{(q)} = \{v : v|_{I_i} \in \mathcal{P}^q(I_i),\, i = 1, ..., m+1\}, \tag{10.25}$$

and the continuous piecewise polynomials on I, defined by

$$V_h^{(q)} = \{v \in W_h^{(q)} : v \in \mathcal{C}(I)\}. \tag{10.26}$$

The basis functions for $W_h^{(q)}$ can be defined in terms of the Lagrange basis, with for example

$$\lambda_{i,0}(x)\frac{x-x_i}{x_{i-1}-x_i} \quad \lambda_{i,1}(x)\frac{x-x_{i-1}}{x_i-x_{i-1}}, \tag{10.27}$$

defining the basis functions for $W_h^{(1)}$, by

$$\phi_{i,j}(x) = \begin{cases} 0, & x \neq [x_{i-1}, x_i], \\ \lambda_{i,j}, & x \in [x_{i-1}, x_i], \end{cases} \tag{10.28}$$

for $i = 1, ..., m+1$, and $j = 0, 1$. For $V_h^{(q)}$ we need to construct continuous basis functions, for example,

$$\phi_i(x) = \begin{cases} 0, & x \neq [x_{i-1}, x_{i+1}], \\ \lambda_{i,1}, & x \in [x_{i-1}, x_i], \\ \lambda_{i+1,0}, & x \in [x_i, x_{i+1}], \end{cases} \tag{10.29}$$

which we also refer to as *hat functions*.

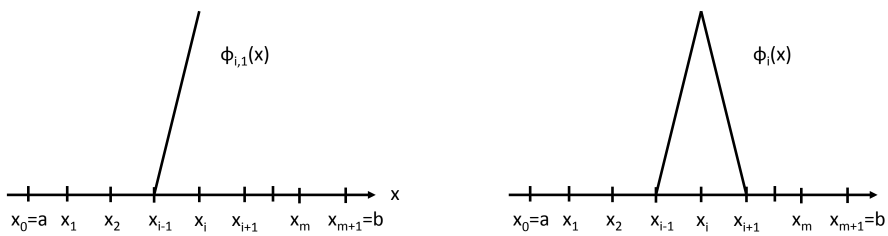

Figure 10.1: Illustration of a *mesh* $\mathcal{T}_h = I_i$, with subintervals $I_j = (x_{i-1}, x_i)$ of length $h_i = x_i - x_{i-1}$, and $\phi_i(x)$ a basis function for $W_h^{(1)}$ (left) and $V_h^{(1)}$ (right).