Chapter 2
Approximation and Interpolation

The present chapter is basically concerned with the approximation of functions. The functions in question may be functions defined on a continuum – typically a finite interval – or functions defined only on a finite set of points. The first instance arises, for example, in the context of special functions (elementary or transcendental) that one wishes to evaluate as a part of a subroutine. Since any such evaluation must be reduced to a finite number of arithmetic operations, we must ultimately approximate the function by means of a polynomial or a rational function. The second instance is frequently encountered in the physical sciences when measurements are taken of a certain physical quantity as a function of some other physical quantity (such as time). In either case one wants to approximate the given function "as well as possible" in terms of other simpler functions.

The general scheme of approximation can be described as follows. We are given the function f to be approximated, along with a class Φ of "approximating functions" φ and a "norm" $\|\cdot\|$ measuring the overall magnitude of functions. We are looking for an approximation $\hat{\varphi} \in \Phi$ of f such that

$$\|f - \hat{\varphi}\| \leq \|f - \varphi\| \quad \text{for all} \quad \varphi \in \Phi. \tag{2.1}$$

The function $\hat{\varphi}$ is called the *best approximation* to f from the class Φ, relative to the norm $\|\cdot\|$.

The class Φ is called a (real) *linear space* if with any two functions φ_1, $\varphi_2 \in \Phi$ it also contains $\varphi_1 + \varphi_2$ and $c\varphi_1$ for any $c \in \mathbb{R}$, hence also any (finite) linear combination of functions $\varphi_i \in \Phi$. Given n "basis functions" $\pi_j \in \Phi$, $j = 1, 2, \ldots, n$, we can define a linear space of finite dimension n by

$$\Phi = \Phi_n = \left\{ \varphi : \ \varphi(t) = \sum_{j=1}^{n} c_j \pi_j(t), \ c_j \in \mathbb{R} \right\}. \tag{2.2}$$

Examples of linear spaces Φ. 1. $\Phi = \mathbb{P}_m$: polynomials of degree $\leq m$. A basis for $\mathbb{P}_m$ is, for example, $\pi_j(t) = t^{j-1}$, $j = 1, 2, \ldots, m+1$, so that $n = m + 1$. Polynomials are the most frequently used "general-purpose" approximants for

W. Gautschi, *Numerical Analysis*, DOI 10.1007/978-0-8176-8259-0_2,

dealing with functions on bounded domains (finite intervals or finite sets of points). One reason is *Weierstrass's theorem*, which states that any continuous function can be approximated on a finite interval as closely as one wishes by a polynomial of sufficiently high degree.

2. $\Phi = \mathbb{S}_m^k(\Delta)$: (polynomial) spline functions of degree m and smoothness class k on the subdivision

$$\Delta: \quad a = t_1 < t_2 < t_3 < \cdots < t_{N-1} < t_N = b$$

of the interval $[a, b]$. These are piecewise polynomials of degree $\leq m$ pieced together at the "joints" $t_2, \ldots, t_{N-1}$ in such a way that all derivatives up to and including the kth are continuous on the whole interval $[a, b]$, including the joints:

$$\mathbb{S}_m^k(\Delta) = \{s \in C^k[a, b]: \ s\big|_{[t_i, t_{i+1}]} \in \mathbb{P}_m, \ i = 1, 2, \ldots, N-1\}.$$

We assume here $0 \leq k < m$; otherwise, we are back to polynomials $\mathbb{P}_m$ (see Ex. 68). We set $k = -1$ if we allow discontinuities at the joints. The dimension of $\mathbb{S}_m^k(\Delta)$ is $n = (m-k) \cdot (N-2) + m + 1$ (see Ex. 71), but to find a basis is a nontrivial task; for $m = 1$, see Sect. 2.3.2.
3. $\Phi = \mathbb{T}_m\,[0, 2\pi]$: trigonometric polynomials of degree $\leq m$ on $[0, 2\pi]$. These are linear combinations of the basic harmonics up to and including the mth one, that is,

$$\pi_k(t) = \cos\,(k-1)t, \ \ k = 1, 2, \ldots, m+1;$$

$$\pi_{m+1+k}(t) = \sin kt, \ \ k = 1, 2, \ldots, m,$$

where now $n = 2m + 1$. Such approximants are a natural choice when the function f to be approximated is periodic with period 2π. (If f has period p, one makes a preliminary change of variables $t \mapsto t \cdot p/2\pi$.)
4. $\Phi = \mathbb{E}_n$: exponential sums. For given distinct $\alpha_j > 0$, one takes $\pi_j(t) = \mathrm{e}^{-\alpha_j t}$, $j = 1, 2, \ldots, n$. Exponential sums are often employed on the half-infinite interval $\mathbb{R}_+$: $0 \leq t < \infty$, especially if one knows that f decays exponentially as $t \to \infty$.

Note that the important class of rational functions,

$$\Phi = \mathbb{R}_{r,s} = \{\varphi: \ \varphi = p/q, \ p \in \mathbb{P}_r, \ q \in \mathbb{P}_s\},$$

is *not* a linear space. (Why not?)

Possible choices of norm – both for continuous and discrete functions – and the type of approximation they generate are summarized in Table 2.1. The continuous case involves an interval $[a, b]$ and a "weight function" $w(t)$ (possibly $w(t) \equiv 1$) defined on $[a, b]$ and positive except for isolated zeros. The discrete case involves a set of N distinct points $t_1, t_2, \ldots, t_N$ along with positive weight factors

Table 2.1 Types of approximation and associated norms

Continuous norm	Approximation	Discrete norm
$\|u\|_\infty = \max_{a\le t\le b} \|u(t)\|$	L_∞	$\|u\|_\infty = \max_{1\le i\le N} \|u(t_i)\|$
	Uniform	
	Chebyshev	
$\|u\|_1 = \int_a^b \|u(t)\|\mathrm{d}t$	L_1	$\|u\|_1 = \sum_{i=1}^N \|u(t_i)\|$
$\|u\|_{1,w} = \int_a^b \|u(t)\|w(t)\mathrm{d}t$	Weighted L_1	$\|u\|_{1,w} = \sum_{i=1}^N w_i\|u(t_i)\|$
$\|u\|_{2,w} = \left(\int_a^b \|u(t)\|^2 w(t)\mathrm{d}t\right)^{\frac{1}{2}}$	Weighted L_2	$\|u\|_{2,w} = \left(\sum_{i=1}^N w_i\|u(t_i)\|^2\right)^{\frac{1}{2}}$
	Least squares	

$w_1, w_2, \ldots, w_N$ (possibly all equal to 1). The interval $[a,b]$ may be unbounded if the weight function w is such that the integral extended over $[a,b]$, which defines the norm, makes sense.

Hence, we may take any one of the norms in Table 2.1 and combine it with any of the preceding linear spaces Φ to arrive at a meaningful best approximation problem (2.1). In the continuous case, the given function f, and the functions φ of the class Φ, of course, must be defined on $[a,b]$ and such that the norm $\|f-\varphi\|$ makes sense. Likewise, f and φ must be defined at the points t_i in the discrete case.

Note that if the best approximant $\hat{\varphi}$ in the discrete case is such that $\|f-\hat{\varphi}\| = 0$, then $\hat{\varphi}(t_i) = f(t_i)$ for $i = 1, 2, \ldots, N$. We then say that $\hat{\varphi}$ *interpolates* f at the points t_i and we refer to this kind of approximation problem as an *interpolation problem*.

The simplest approximation problems are the least squares problem and the interpolation problem, and the easiest space Φ to work with the space of polynomials of given degree. These are indeed the problems we concentrate on in this chapter. In the case of the least squares problem, however, we admit general linear spaces Φ of approximants, and also in the case of the interpolation problem, we include polynomial splines in addition to straight polynomials.

Before we start with the least squares problem, we introduce a notational device that allows us to treat the continuous and the discrete case simultaneously. We define, in the continuous case,

$$\lambda(t) = \begin{cases} 0 \ \text{ if } \ t < a \ \ (\text{whenever } -\infty < a), \\ \displaystyle\int_a^t w(\tau)\mathrm{d}\tau \ \text{ if } \ a \le t \le b, \\ \displaystyle\int_a^b w(\tau)\mathrm{d}\tau \ \text{ if } \ t > b \ \ (\text{whenever } b < \infty). \end{cases} \tag{2.3}$$

Then we can write, for any (say, continuous) function u,

$$\int_{\mathbb{R}} u(t)\mathrm{d}\lambda(t) = \int_a^b u(t)w(t)\mathrm{d}t, \tag{2.4}$$

since $\mathrm{d}\lambda(t) \equiv 0$ "outside" $[a, b]$, and $\mathrm{d}\lambda(t) = w(t)\mathrm{d}t$ inside. We call $\mathrm{d}\lambda$ a *continuous* (positive) *measure*. The *discrete measure* (also called "Dirac measure") associated with the point set $\{t_1, t_2, \ldots, t_N\}$ is a measure $\mathrm{d}\lambda$ that is nonzero only at the points t_i and has the value w_i there. Thus, in this case,

$$\int_{\mathbb{R}} u(t)\mathrm{d}\lambda(t) = \sum_{i=1}^{N} w_i u(t_i). \tag{2.5}$$

(A more precise definition can be given in terms of Stieltjes integrals, if we define $\lambda(t)$ to be a *step function* having jump w_i at t_i.) In particular, we can define the L_2 norm as

$$\|u\|_{2,\mathrm{d}\lambda} = \left(\int_{\mathbb{R}} |u(t)|^2 \mathrm{d}\lambda(t)\right)^{\frac{1}{2}}, \tag{2.6}$$

and obtain the continuous or the discrete norm depending on whether λ is taken to be as in (2.3), or a step function, as in (2.5).

We call the *support* of $\mathrm{d}\lambda$ – and denote it by $\operatorname{supp}\mathrm{d}\lambda$ – the interval $[a, b]$ in the continuous case (assuming w positive on $[a, b]$ except for isolated zeros), and the set $\{t_1, t_2, \ldots, t_N\}$ in the discrete case. We say that the set of functions $\pi_j(t)$ in (2.2) is *linearly independent* on the support of $\mathrm{d}\lambda$ if

$$\sum_{j=1}^{n} c_j \pi_j(t) \equiv 0 \ \text{ for all } \ t \in \operatorname{supp}\mathrm{d}\lambda \ \text{ implies } \ c_1 = c_2 = \cdots = c_n = 0. \tag{2.7}$$

Example: the powers $\pi_j(t) = t^{j-1}$, $j = 1, 2, \ldots, n$.

Here $\sum_{j=1}^{n} c_j \pi_j(t) = p_{n-1}(t)$ is a polynomial of degree $\leq n-1$. Suppose, first, that $\operatorname{supp}\mathrm{d}\lambda = [a, b]$. Then the identity in (2.7) says that $p_{n-1}(t) \equiv 0$ on $[a, b]$. Clearly, this implies $c_1 = c_2 = \cdots = c_n = 0$, so that the powers are linearly independent on $\operatorname{supp}\mathrm{d}\lambda = [a, b]$. If, on the other hand, $\operatorname{supp}\mathrm{d}\lambda = \{t_1, t_2, \ldots, t_N\}$, then the premise in (2.7) says that $p_{n-1}(t_i) = 0$, $i = 1, 2, \ldots, N$; that is, p_{n-1} has N distinct zeros t_i. This implies $p_{n-1} \equiv 0$ only if $N \geq n$. Otherwise, $p_{n-1}(t) = \prod_{i=1}^{N}(t - t_i) \in \mathbb{P}_{n-1}$ would satisfy $p_{n-1}(t_i) = 0$, $i = 1, 2, \ldots, N$, without being identically zero. Thus, we have linear independence on $\operatorname{supp}\mathrm{d}\lambda = \{t_1, t_2, \ldots, t_N\}$ if and only if $N \geq n$.

2.1 Least Squares Approximation

We specialize the best approximation problem (2.1) by taking as norm the L_2 norm

$$\|u\|_{2,\mathrm{d}\lambda} = \left(\int_{\mathbb{R}} |u(t)|^2 \mathrm{d}\lambda(t) \right)^{\frac{1}{2}}, \tag{2.8}$$

where $\mathrm{d}\lambda$ is either a continuous measure (cf. (2.3)) or a discrete measure (cf. (2.5)), and by using approximants φ from an n-dimensional linear space

$$\Phi = \Phi_n = \left\{ \varphi : \;\; \varphi(t) = \sum_{j=1}^{n} c_j \pi_j(t),\; c_j \in \mathbb{R} \right\}. \tag{2.9}$$

Here the basis functions π_j are assumed linearly independent on $\operatorname{supp} \mathrm{d}\lambda$ (cf. (2.7)). We furthermore assume, of course, that the integral in (2.8) is meaningful whenever $u = \pi_j$ or $u = f$, the given function to be approximated.

The solution of the least squares problem is most easily expressed in terms of orthogonal systems π_j relative to an appropriate inner product. We therefore begin with a discussion of inner products.

2.1.1 Inner Products

Given a continuous or discrete measure $\mathrm{d}\lambda$, as introduced earlier, and given any two functions u, v having a finite norm (2.8), we can define the *inner product*

$$(u, v) = \int_{\mathbb{R}} u(t)v(t)\mathrm{d}\lambda(t). \tag{2.10}$$

(Schwarz's inequality $|(u,v)| \le \|u\|_{2,\mathrm{d}\lambda} \cdot \|v\|_{2,\mathrm{d}\lambda}$, cf. Ex. 6, tells us that the integral in (2.10) is well defined.) The inner product (2.10) has the following obvious (but useful) properties:

1. symmetry: $(u, v) = (v, u)$;
2. homogeneity: $(\alpha u, v) = \alpha(u, v)$, $\alpha \in \mathbb{R}$;
3. additivity: $(u + v, w) = (u, w) + (v, w)$; and
4. positive definiteness: $(u, u) \ge 0$, with equality holding if and only if $u \equiv 0$ on $\operatorname{supp} \mathrm{d}\lambda$.

Homogeneity and additivity together give *linearity*,

$$(\alpha_1 u_1 + \alpha_2 u_2, v) = \alpha_1(u_1, v) + \alpha_2(u_2, v) \tag{2.11}$$

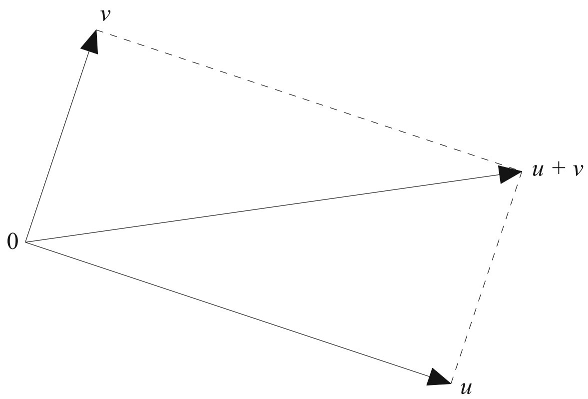


Fig. 2.1 Orthogonal vectors and their sum

in the first variable and, by symmetry, also in the second. Moreover, (2.11) easily extends to linear combinations of arbitrary finite length. Note also that

$$\|u\|^2_{2,\,\mathrm{d}\lambda} = (u, u). \tag{2.12}$$

We say that u and v are *orthogonal* if

$$(u, v) = 0. \tag{2.13}$$

This is always trivially true if either u or v vanishes identically on supp $\mathrm{d}\lambda$.

It is now a simple exercise, for example, to prove the *Theorem of Pythagoras*:

$$\text{if } (u, v) = 0, \text{ then } \|u + v\|^2 = \|u\|^2 + \|v\|^2, \tag{2.14}$$

where $\|\cdot\| = \|\cdot\|_{2,\mathrm{d}\lambda}$. (From now on we use this abbreviated notation for the norm.) Indeed,

$$\begin{aligned}\|u + v\|^2 &= (u + v, u + v) = (u, u) + (u, v) + (v, u) + (v, v)\\ &= \|u\|^2 + 2(u, v) + \|v\|^2 = \|u\|^2 + \|v\|^2,\end{aligned}$$

where the first equality is a definition, the second follows from additivity, the third from symmetry, and the last from orthogonality. Interpreting functions u, v as "vectors," we can picture the configuration of u, v (orthogonal) and $u + v$ as in Fig. 2.1.

More generally, we may consider an *orthogonal systems* $\{u_k\}_{k=1}^n$:

$$\begin{aligned}&(u_i, u_j) = 0 \text{ if } i \neq j, \ \ u_k \not\equiv 0 \text{ on supp } \mathrm{d}\lambda;\\ &i, j = 1, 2, \ldots, n; \ \ k = 1, 2, \ldots, n.\end{aligned} \tag{2.15}$$

For such a system we have the *Generalized Theorem of Pythagoras*,

$$\left\|\sum_{k=1}^{n} \alpha_k u_k\right\|^2 = \sum_{k=1}^{n} |\alpha_k|^2 \|u_k\|^2. \tag{2.16}$$

The proof is essentially the same as before. An important consequence of (2.16) is that *every orthogonal system is linearly independent* on the support of $\mathrm{d}\lambda$. Indeed, if the left-hand side of (2.16) vanishes, then so does the right-hand side, and this, since $\|u_k\|^2 > 0$ by assumption, implies $\alpha_1 = \alpha_2 = \cdots = \alpha_n = 0$.

2.1.2 The Normal Equations

We are now in a position to solve the least squares approximation problem. By (2.12), we can write the L_2 error, or rather its square, in the form:

$$E^2[\varphi] := \|\varphi - f\|^2 = (\varphi - f, \varphi - f) = (\varphi, \varphi) - 2(\varphi, f) + (f, f).$$

Inserting φ here from (2.9) gives

$$E^2[\varphi] = \int_{\mathbb{R}} \left(\sum_{j=1}^{n} c_j \pi_j(t)\right)^2 \mathrm{d}\lambda(t) - 2\int_{\mathbb{R}} \left(\sum_{j=1}^{n} c_j \pi_j(t)\right) f(t)\mathrm{d}\lambda(t) + \int_{\mathbb{R}} f^2(t)\mathrm{d}\lambda(t). \tag{2.17}$$

The squared L_2 error, therefore, is a quadratic function of the coefficients c_1, $c_2, \ldots, c_n$ of φ. The problem of best L_2 approximation thus amounts to minimizing a quadratic function of n variables. This is a standard problem of calculus and is solved by setting all partial derivatives equal to zero. This yields a system of *linear* algebraic equations. Indeed, differentiating partially with respect to c_i under the integral sign in (2.17) gives

$$\frac{\partial}{\partial c_i} E^2[\varphi] = 2\int_{\mathbb{R}} \left(\sum_{j=1}^{n} c_j \pi_j(t)\right) \pi_i(t)\mathrm{d}\lambda(t) - 2\int_{\mathbb{R}} \pi_i(t) f(t)\mathrm{d}\lambda(t),$$

and setting this equal to zero, interchanging integration and summation in the process, we get

$$\sum_{j=1}^{n} (\pi_i, \pi_j) c_j = (\pi_i, f), \quad i = 1, 2, \ldots, n. \tag{2.18}$$

These are called the *normal equations* for the least squares problem. They form a linear system of the form

$$\boldsymbol{A}\boldsymbol{c} = \boldsymbol{b}, \tag{2.19}$$

where the matrix $\boldsymbol{A}$ and the vector $\boldsymbol{b}$ have elements

$$\boldsymbol{A} = [a_{ij}],\ a_{ij} = (\pi_i, \pi_j);\ \ \boldsymbol{b} = [b_i],\ b_i = (\pi_i, f). \tag{2.20}$$

By symmetry of the inner product, $\boldsymbol{A}$ is a symmetric matrix. Moreover, $\boldsymbol{A}$ is positive definite; that is,

$$\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}x_i x_j > 0 \ \text{ if } \ \boldsymbol{x} \neq [0, 0, \ldots, 0]^{\mathrm{T}}. \tag{2.21}$$

The quadratic function in (2.21) is called a *quadratic form* (since it is homogeneous of degree 2). Positive definiteness of $\boldsymbol{A}$ thus says that the quadratic form whose coefficients are the elements of $\boldsymbol{A}$ is always nonnegative, and zero only if all variables x_i vanish.

To prove (2.21), all we have to do is insert the definition of the a_{ij} and use the elementary properties 1–4 of the inner product:

$$\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} = \sum_{i=1}^{n}\sum_{j=1}^{n} x_i x_j (\pi_i, \pi_j) = \sum_{i=1}^{n}\sum_{j=1}^{n} (x_i\pi_i, x_j\pi_j) = \left\| \sum_{i=1}^{n} x_i\pi_i \right\|^2.$$

This clearly is nonnegative. It is zero only if $\sum_{i=1}^{n} x_i\pi_i \equiv 0$ on $\operatorname{supp} \mathrm{d}\lambda$, which, by the assumption of linear independence of the π_i, implies $x_1 = x_2 = \cdots = x_n = 0$.

Now it is a well-known fact of linear algebra that a symmetric positive definite matrix $\boldsymbol{A}$ is nonsingular. Indeed, its determinant, as well as all its leading principal minor determinants, are strictly positive. It follows that the system (2.18) of normal equations has a unique solution. Does this solution correspond to a minimum of $E[\varphi]$ in (2.17)? Calculus tells us that for this to be the case, the Hessian matrix $\boldsymbol{H} = [\partial^2 E^2/\partial c_i \partial c_j]$ has to be positive definite. But $\boldsymbol{H} = 2\boldsymbol{A}$, since E^2 is a quadratic function. Therefore, $\boldsymbol{H}$, with $\boldsymbol{A}$, is indeed positive definite, and the solution of the normal equations gives us the desired minimum. The least squares approximation problem thus has a unique solution, given by

$$\hat{\varphi}(t) = \sum_{j=1}^{n} \hat{c}_j \pi_j(t), \tag{2.22}$$

where $\hat{\boldsymbol{c}} = [\hat{c}_1, \hat{c}_2, \ldots, \hat{c}_n]^{\mathrm{T}}$ is the solution vector of the normal equation (2.18).

This completely settles the least squares approximation problem in theory. How about in practice?

Assuming a general set of (linearly independent) basis functions, we can see the following possible difficulties.

1. The system (2.18) may be *ill-conditioned*. A simple example is provided by $\operatorname{supp} \mathrm{d}\lambda = [0, 1]$, $\mathrm{d}\lambda(t) = \mathrm{d}t$ on [0,1], and $\pi_j(t) = t^{j-1}$, $j = 1, 2, \ldots, n$. Then

$$(\pi_i, \pi_j) = \int_0^1 t^{i+j-2}\,\mathrm{d}t = \frac{1}{i+j-1},\ \ i, j = 1, 2, \ldots, n;$$

that is, the matrix $\boldsymbol{A}$ in (2.18) is precisely the Hilbert matrix (cf. Chap. 1, (1.60)). The resulting severe ill-conditioning of the normal equations in this example is entirely due to an unfortunate choice of basis functions – the powers. These become almost linearly dependent, more so the larger the exponent (cf. Ex. 38). Another source of degradation lies in the element $b_j = \int_0^1 \pi_j(t) f(t)\mathrm{d}t$ of the right-hand vector $\boldsymbol{b}$ in (2.18). When j is large, the power $\pi_j = t^{j-1}$ behaves very much like a discontinuous function on [0,1]: it is practically zero for much of the interval until it shoots up to the value 1 at the right endpoint. This has the unfortunate consequence that a good deal of information about f is lost when one forms the integral defining b_j. A polynomial π_j that oscillates rapidly on [0,1] would seem to be preferable from this point of view, since it would "engage" the function f more vigorously over all of the interval [0,1].

2. The second disadvantage is the fact that all coefficients $\hat{c}_j$ in (2.22) depend on n; that is, $\hat{c}_j = \hat{c}_j^{(n)}$, $j = 1, 2, \ldots, n$. Increasing n, for example, will give an enlarged system of normal equations with a completely new solution vector. We refer to this as the *nonpermanence* of the coefficients $\hat{c}_j$.

Both defects 1 and 2 can be eliminated (or at least attenuated in the case of 1) in one stroke: select for the basis functions π_j an orthogonal system,

$$(\pi_i, \pi_j) = 0 \ \text{ if } \ i \neq j; \ \ (\pi_j, \pi_j) = \|\pi_j\|^2 > 0. \tag{2.23}$$

Then the system of normal equations becomes diagonal and is solved immediately by

$$\hat{c}_j = \frac{(\pi_j, f)}{(\pi_j, \pi_j)}, \quad j = 1, 2, \ldots, n. \tag{2.24}$$

Clearly, each of these coefficients $\hat{c}_j$ is independent of n, and once computed, remains the same for any larger n. We now have *permanence* of the coefficients. Also, we do not have to go through the trouble of solving a linear system of equations, but instead can use the formula (2.24) directly. This does not mean that there are no numerical problems associated with (2.24). Indeed, it is typical that the denominators $\|\pi_j\|^2$ in (2.24) decrease rapidly with increasing j, whereas the integrand in the numerator (or the individual terms in the case of a discrete inner product) are of the same magnitude as f. Yet the coefficients $\hat{c}_j$ also are expected to decrease rapidly. Therefore, cancellation errors must occur when one computes the inner product in the numerator. The cancellation problem can be alleviated somewhat by computing $\hat{c}_j$ in the alternative form

$$\hat{c}_j = \frac{1}{(\pi_j, \pi_j)} \left(f - \sum_{k=1}^{j-1} \hat{c}_k \pi_k, \pi_j \right), \quad j = 1, 2, \ldots, n, \tag{2.25}$$

where the empty sum (when $j = 1$) is taken to be zero, as usual. Clearly, by orthogonality of the π_j, (2.25) is equivalent to (2.24) mathematically, but not necessarily numerically.

An algorithm for computing $\hat{c}_j$ from (2.25), and at the same time $\hat{\varphi}(t)$, is as follows:

$$\begin{aligned}
&s_0 = 0,\\
&\text{for } j = 1, 2, \dots, n \text{ do}\\
&\quad\left[\begin{array}{l} \hat{c}_j = \dfrac{1}{\|\pi_j\|^2}(f - s_{j-1}, \pi_j) \\ s_j = s_{j-1} + \hat{c}_j \pi_j(t). \end{array}\right.
\end{aligned}$$

This produces the coefficients $\hat{c}_1, \hat{c}_2, \dots, \hat{c}_n$ as well as $\hat{\varphi}(t) = s_n$.

Any system $\{\hat{\pi}_j\}$ that is linearly independent on the support of $d\lambda$ can be orthogonalized (with respect to the measure $d\lambda$) by a device known as the *Gram*[1]–Schmidt [2]*procedure*. One takes

$$\pi_1 = \hat{\pi}_1$$

and, for $j = 2, 3, \dots$, recursively forms

$$\pi_j = \hat{\pi}_j - \sum_{k=1}^{j-1} c_k \pi_k, \quad c_k = \frac{(\hat{\pi}_j, \pi_k)}{(\pi_k, \pi_k)}.$$

Then each π_j so determined is orthogonal to all preceding ones.

2.1.3 Least Squares Error; Convergence

We have seen in Sect. 2.1.2 that if the class $\Phi = \Phi_n$ consists of n functions π_j, $j = 1, 2, \dots, n$, that are linearly independent on the support of some measure $d\lambda$, then the least squares problem for this measure,

$$\min_{\varphi \in \Phi_n} \|f - \varphi\|_{2,d\lambda} = \|f - \hat{\varphi}\|_{2,d\lambda}, \tag{2.26}$$

[1]Jórgen Pedersen Gram (1850–1916) was a farmer's son who studied at the University of Copenhagen. After graduation, he entered an insurance company as computer assistant and, moving up the ranks, eventually became its director. He was interested in series expansions of special functions and also contributed to Chebyshev and least squares approximation. The "Gram determinant" was introduced by him in connection with his study of linear independence.

[2]Erhard Schmidt (1876–1959), a student of Hilbert, became a prominent member of the Berlin School of Mathematics, where he founded the Institute of Applied Mathematics. He is considered one of the originators of Functional Analysis, having contributed substantially to the theory of Hilbert spaces. His work on linear and nonlinear integral equations is of lasting interest, as is his contribution to linear algebraic systems of infinite dimension. He is also known for his proof of the Jordan curve theorem. His procedure of orthogonalization was published in 1907 and today also carries the name of Gram. It was known, however, already to Laplace.

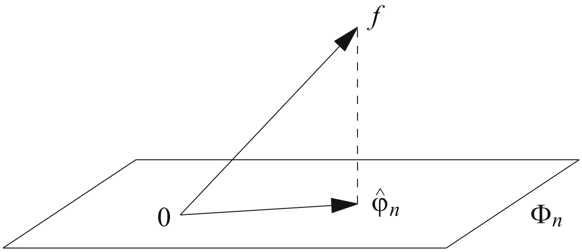


Fig. 2.2 Least squares approximation as orthogonal projection

has a unique solution $\hat{\varphi} = \hat{\varphi}_n$ given by (2.22). There are many ways we can select a basis π_j in Φ_n and, therefore, many ways the solution $\hat{\varphi}_n$ can be represented. Nevertheless, it is always one and the same function. The least squares error – the quantity on the right-hand side of (2.26) – therefore is independent of the choice of basis functions (although the calculation of the least squares solution, as mentioned previously, is not). In studying this error, we may thus assume, without restricting generality, that the basis π_j is an orthogonal system. (Every linearly independent system can be orthogonalized by the Gram–Schmidt orthogonalization procedure; cf. Sect. 2.1.2.) We then have (cf. (2.24))

$$\hat{\varphi}_n(t) = \sum_{j=1}^{n} \hat{c}_j \pi_j(t), \quad \hat{c}_j = \frac{(\pi_j, f)}{(\pi_j, \pi_j)}. \tag{2.27}$$

We first note that the error $f - \hat{\varphi}_n$ is orthogonal to the space Φ_n; that is,

$$(f - \hat{\varphi}_n, \varphi) = 0 \ \text{ for all } \ \varphi \in \Phi_n, \tag{2.28}$$

where the inner product is the one in (2.10). Since φ is a linear combination of the π_k, it suffices to show (2.28) for each $\varphi = \pi_k$, $k = 1, 2, \ldots, n$. Inserting $\hat{\varphi}_n$ from (2.27) in the left-hand side of (2.28), and using orthogonality, we find indeed

$$(f - \hat{\varphi}_n, \pi_k) = \left(f - \sum_{j=1}^{n} \hat{c}_j \pi_j, \pi_k \right) = (f, \pi_k) - \hat{c}_k(\pi_k, \pi_k) = 0,$$

the last equation following from the formula for $\hat{c}_k$ in (2.27). The result (2.28) has a simple geometric interpretation. If we picture functions as vectors, and the space Φ_n as a plane, then for any f that "sticks out" of the plane Φ_n, the least squares approximant $\hat{\varphi}_n$ is the *orthogonal projection* of f onto Φ_n; see Fig. 2.2.

In particular, choosing $\varphi = \hat{\varphi}_n$ in (2.28), we get

$$(f - \hat{\varphi}_n, \hat{\varphi}_n) = 0$$

and, therefore, since $f = (f - \hat{\varphi}_n) + \hat{\varphi}_n$, by the Theorem of Pythagoras (cf. (2.14)) and its generalization (cf. (2.16)),

$$\begin{aligned}
\|f\|^2 &= \|f - \hat{\varphi}_n\|^2 + \|\hat{\varphi}_n\|^2 \\
&= \|f - \hat{\varphi}_n\|^2 + \left\| \sum_{j=1}^{n} \hat{c}_j \pi_j \right\|^2 \\
&= \|f - \hat{\varphi}_n\|^2 + \sum_{j=1}^{n} |\hat{c}_j|^2 \|\pi_j\|^2.
\end{aligned}$$

Solving for the first term on the right-hand side, we get

$$\|f - \hat{\varphi}_n\| = \left\{ \|f\|^2 - \sum_{j=1}^{n} |\hat{c}_j|^2 \|\pi_j\|^2 \right\}^{\frac{1}{2}}, \quad \hat{c}_j = \frac{(\pi_j, f)}{(\pi_j, \pi_j)}. \tag{2.29}$$

Note that the expression in braces must necessarily be nonnegative.

The formula (2.29) for the error is interesting theoretically, but of limited practical use. Note, indeed, that as the error approaches the level of the machine precision eps, computing the error from the right-hand side of (2.29) cannot produce anything smaller than $\sqrt{\text{eps}}$ because of inevitable rounding errors committed during the subtraction in the radicand. (They may even produce a negative result for the radicand.) Using instead the definition,

$$\|f - \hat{\varphi}_n\| = \left\{ \int_{\mathbb{R}} [f(t) - \hat{\varphi}_n(t)]^2 \mathrm{d}\lambda(t) \right\}^{\frac{1}{2}},$$

along, perhaps, with a suitable (positive) quadrature rule (cf. Chap. 3, Sect. 3.2), is guaranteed to produce a nonnegative result that may potentially be as small as $O(\text{eps})$.

If we are now given a sequence of linear spaces Φ_n, $n = 1, 2, 3, \ldots$, as defined in (2.2), then clearly

$$\|f - \hat{\varphi}_1\| \geq \|f - \hat{\varphi}_2\| \geq \|f - \hat{\varphi}_3\| \geq \cdots,$$

which follows not only from (2.29), but more directly from the fact that $\Phi_1 \subset \Phi_2 \subset \Phi_3 \subset \cdots$. If there are infinitely many such spaces, then the sequence of L_2 errors, being monotonically decreasing, must converge to a limit. Is this limit zero? If so, we say that the least squares approximation process *converges* (in the mean) as $n \to \infty$. It is obvious from (2.29) that a necessary and sufficient condition for this is

$$\sum_{j=1}^{\infty} |\hat{c}_j|^2 \|\pi_j\|^2 = \|f\|^2. \tag{2.30}$$

An equivalent way of stating convergence is as follows: given any f with $\|f\| < \infty$, that is, any f in the $L_{2,\mathrm{d}\lambda}$ space, and given any $\varepsilon > 0$, no matter how small, there exists an integer $n = n_\varepsilon$ and a function $\varphi^* \in \Phi_n$ such that $\|f - \varphi^*\| \le \varepsilon$. A class of spaces Φ_n having this property is said to be *complete* with respect to the norm $\|\cdot\| = \|\cdot\|_{2,\mathrm{d}\lambda}$. One therefore calls (2.30) also the *completeness relation*.

For a finite interval $[a, b]$, one can define completeness of $\{\Phi_n\}$ also for the uniform norm $\|\cdot\| = \|\cdot\|_\infty$ on $[a, b]$. One then assumes $f \in C[a, b]$ and also $\pi_j \in C[a, b]$ for all basis functions in all classes Φ_n, and one calls $\{\Phi_n\}$ complete in the norm $\|\cdot\|_\infty$ if for any $f \in C[a, b]$ and any $\varepsilon > 0$ there is an $n = n_\varepsilon$ and a $\varphi^* \in \Phi_n$ such that $\|f - \varphi^*\|_\infty \le \varepsilon$. It is easy to see that completeness of $\{\Phi_n\}$ in the norm $\|\cdot\|_\infty$ (on $[a, b]$) implies completeness of $\{\Phi_n\}$ in the L_2 norm $\|\cdot\|_{2,\mathrm{d}\lambda}$, where $\operatorname{supp} \mathrm{d}\lambda = [a, b]$, and hence convergence of the least squares approximation process. Indeed, let $\varepsilon > 0$ be arbitrary and let n and $\varphi^* \in \Phi_n$ be such that

$$\|f - \varphi^*\|_\infty \le \frac{\varepsilon}{\left(\int_{\mathbb{R}} \mathrm{d}\lambda(t)\right)^{\frac{1}{2}}}.$$

This is possible by assumption. Then

$$\begin{aligned}\|f - \varphi^*\|_{2,\mathrm{d}\lambda} &= \left(\int_{\mathbb{R}} [f(t) - \varphi^*(t)]^2 \mathrm{d}\lambda(t)\right)^{\frac{1}{2}} \\ &\le \|f - \varphi^*\|_\infty \left(\int_{\mathbb{R}} \mathrm{d}\lambda(t)\right)^{\frac{1}{2}} \le \frac{\varepsilon}{\left(\int_{\mathbb{R}} \mathrm{d}\lambda(t)\right)^{\frac{1}{2}}} \left(\int_{\mathbb{R}} \mathrm{d}\lambda(t)\right)^{\frac{1}{2}} = \varepsilon,\end{aligned}$$

as claimed.

Example: $\Phi_n = \mathbb{P}_{n-1}$.

Here completeness of $\{\Phi_n\}$ in the norm $\|\cdot\|_\infty$ (on a finite interval $[a, b]$) is a consequence of Weierstrass's Approximation Theorem. Thus, polynomial least squares approximation on a finite interval always converges (in the mean).

2.1.4 Examples of Orthogonal Systems

There are many orthogonal systems in use. The prototype of them all is the system of trigonometric functions known from Fourier analysis. Other widely used systems involve algebraic polynomials. We restrict ourselves here to these two particular examples of orthogonal systems.

1. *Trigonometric functions*: $1,\ \cos t,\ \cos 2t,\ \cos 3t\ ,\ldots,\ \sin t,\ \sin 2t,\ \sin 3t,\ \ldots$. These are the basic harmonics; they are mutually orthogonal on the interval $[0, 2\pi]$ with respect to the equally weighted measure on $[0,2\pi]$,

$$d\lambda(t) = \begin{cases} dt & \text{on } [0, 2\pi], \\ 0 & \text{otherwise.} \end{cases} \tag{2.31}$$

We verify this for the sine functions: for $k, \ell = 1, 2, 3, \ldots$ we have

$$\int_0^{2\pi} \sin kt \cdot \sin \ell t \, dt = -\frac{1}{2} \int_0^{2\pi} [\cos(k+\ell)t - \cos(k-\ell)t] \, dt.$$

The right-hand side is equal to

$$-\frac{1}{2}\left[\frac{\sin(k+\ell)t}{k+\ell} - \frac{\sin(k-\ell)t}{k-\ell}\right]_0^{2\pi} = 0,$$

when $k \neq \ell$, and equal to π otherwise. Thus,

$$\int_0^{2\pi} \sin kt \cdot \sin \ell t \, dt = \begin{cases} 0 & \text{if} \quad k \neq \ell, \\ \pi & \text{if} \quad k = \ell, \end{cases} \quad k, \ell = 1, 2, 3, \ldots . \tag{2.32}$$

Similarly, one shows that

$$\int_0^{2\pi} \cos kt \cdot \cos \ell t \, dt = \begin{cases} 0 & \text{if} \quad k \neq \ell, \\ 2\pi & \text{if} \quad k = \ell = 0, \\ \pi & \text{if} \quad k = \ell > 0, \end{cases} \quad k, \ell = 0, 1, 2, \ldots \tag{2.33}$$

and

$$\int_0^{2\pi} \sin kt \cdot \cos \ell t \, dt = 0, \quad k = 1, 2, 3, \ldots, \quad \ell = 0, 1, 2, \ldots . \tag{2.34}$$

The theory of *Fourier series* is concerned with the expansion of a given 2π-periodic function in terms of these trigonometric functions,

$$f(t) = \sum_{k=0}^{\infty} a_k \cos kt + \sum_{k=1}^{\infty} b_k \sin kt. \tag{2.35}$$

Using (2.32)–(2.34), one formally obtains

$$a_0 = \frac{1}{2\pi}\int_0^{2\pi} f(t)\,\mathrm{d}t,\quad a_k = \frac{1}{\pi}\int_0^{2\pi} f(t)\cos kt\,\mathrm{d}t,\quad k = 1,2,\dots,$$
$$b_k = \frac{1}{\pi}\int_0^{2\pi} f(t)\sin kt\,\mathrm{d}t,\quad k = 1,2,\dots, \tag{2.36}$$

which are known as *Fourier coefficients* of f. They are precisely the coefficients (2.24) for the system π_j consisting of our trigonometric functions. By extension, one therefore calls the coefficients $\hat{c}_j$ in (2.24), for *any* orthogonal system π_j, the Fourier coefficients of f relative to this system. In particular, we now recognize the truncated Fourier series (the series on the right-hand side of (2.35) truncated at $k = m$, with a_k, b_k given by (2.36)) as the best L_2 approximation to f from the class of trigonometric polynomials of degree $\leq m$ relative to the norm (cf. (2.31))

$$\|u\|_2 = \left(\int_0^{2\pi} |u(t)|^2 \mathrm{d}t\right)^{\frac{1}{2}}.$$

2. *Orthogonal polynomials*: given a measure $\mathrm{d}\lambda$ as introduced in (2.3)–(2.5), we know from the example immediately following (2.7) that any finite number of consecutive powers $1, t, t^2, \dots$ are linearly independent on $[a,b]$, if $\operatorname{supp}\mathrm{d}\lambda = [a,b]$, whereas the finite set $1, t, \dots, t^{N-1}$ is linearly independent on $\operatorname{supp}\mathrm{d}\lambda = \{t_1, t_2, \dots, t_N\}$. Since a linearly independent set can be orthogonalized by Gram–Schmidt (cf. Sect. 2.1.2), any measure $\mathrm{d}\lambda$ of the type considered generates a unique set of (monic) polynomials $\pi_j(t) = \pi_j(t;\mathrm{d}\lambda)$, $j = 0,1,2,\dots,$ satisfying

$$\deg \pi_j = j,\quad j = 0,1,2,\dots,$$
$$\int_{\mathbb{R}} \pi_k(t)\pi_\ell(t)\mathrm{d}\lambda(t) = 0 \ \text{ if } k \neq \ell. \tag{2.37}$$

These are called *orthogonal polynomials* relative to the measure $\mathrm{d}\lambda$. (We slightly deviate from the notation in Sects. 2.1.2 and 2.1.3 by letting the index j start from zero.) The set π_j is infinite if $\operatorname{supp}\mathrm{d}\lambda = [a,b]$, and consists of exactly N polynomials $\pi_0, \pi_1, \dots, \pi_{N-1}$ if $\operatorname{supp}\mathrm{d}\lambda = \{t_1, t_2, \dots, t_N\}$. The latter are referred to as *discrete orthogonal polynomials.*

It is an important fact that three consecutive orthogonal polynomials are linearly related. Specifically, there are real constants $\alpha_k = \alpha_k(\mathrm{d}\lambda)$ and positive constants $\beta_k = \beta_k(\mathrm{d}\lambda)$ (depending on the measure $\mathrm{d}\lambda$) such that

$$\pi_{k+1}(t) = (t - \alpha_k)\pi_k(t) - \beta_k \pi_{k-1}(t),\quad k = 0,1,2,\dots,$$
$$\pi_{-1}(t) = 0,\quad \pi_0(t) = 1. \tag{2.38}$$

(It is understood that (2.38) holds for all integers $k \geq 0$ if $\operatorname{supp} d\lambda = [a, b]$, and only for $0 \leq k < N - 1$ if $\operatorname{supp} d\lambda = \{t_1, t_2, \ldots, t_N\}$.)

To prove (2.38) and, at the same time identify the coefficients α_k, β_k, we note that

$$\pi_{k+1}(t) - t\pi_k(t)$$

is a polynomial of degree $\leq k$, since the leading terms cancel (the polynomials π_j are assumed monic). Since an orthogonal system is linearly independent (cf. the remark after (2.16)), we can express this polynomial as a linear combination of π_0, $\pi_1, \ldots, \pi_k$. We choose to write this linear combination in the form:

$$\pi_{k+1}(t) - t\pi_k(t) = -\alpha_k \pi_k(t) - \beta_k \pi_{k-1}(t) + \sum_{j=0}^{k-2} \gamma_{k,j} \pi_j(t) \tag{2.39}$$

(with the understanding that empty sums are zero). Now multiply both sides of (2.39) by π_k in the sense of the inner product $(\cdot\ ,\ \cdot)$ defined in (2.10). By orthogonality, this gives $(-t\pi_k, \pi_k) = -\alpha_k(\pi_k, \pi_k)$; that is,

$$\alpha_k = \frac{(t\pi_k, \pi_k)}{(\pi_k, \pi_k)}, \quad k = 0, 1, 2, \ldots. \tag{2.40}$$

Similarly, forming the inner product of (2.39) with π_{k-1} gives $(-t\pi_k, \pi_{k-1}) = -\beta_k(\pi_{k-1}, \pi_{k-1})$. Since $(t\pi_k, \pi_{k-1}) = (\pi_k, t\pi_{k-1})$ and $t\pi_{k-1}$ differs from π_k by a polynomial of degree $< k$, we obtain by orthogonality $(t\pi_k, \pi_{k-1}) = (\pi_k, \pi_k)$; hence

$$\beta_k = \frac{(\pi_k, \pi_k)}{(\pi_{k-1}, \pi_{k-1})}, \quad k = 1, 2, \ldots. \tag{2.41}$$

Finally, multiplication of (2.39) by π_ℓ, $\ell < k - 1$, yields

$$\gamma_{k,\ell} = 0, \quad \ell = 0, 1, \ldots, k - 2. \tag{2.42}$$

Solving (2.39) for π_{k+1} then establishes (2.38), with α_k, β_k defined by (2.40) and (2.41), respectively. Clearly, $\beta_k > 0$. By convention, $\beta_0 = \int_{\mathbb{R}} d\lambda(t) = \int_{\mathbb{R}} \pi_0^2(t) d\lambda(t)$.

The recursion (2.38) provides us with a practical scheme of generating orthogonal polynomials. Indeed, since $\pi_0 = 1$, we can compute α_0 by (2.40) with $k = 0$. This allows us to compute $\pi_1(t)$ for any t, using (2.38) with $k = 0$. Knowing π_0, π_1, we can go back to (2.40) and (2.41) and compute, respectively, α_1 and β_1. This gives us access to π_2 via (2.38) with $k = 1$. Proceeding in this fashion, using alternately (2.40), (2.41), and (2.38), we can generate as many orthogonal polynomials as are desired. This procedure – called *Stieltjes's procedure* – is particularly well suited for discrete orthogonal polynomials, since the inner product is then a finite sum, $(u, v) = \sum_{i=1}^{N} w_i u(t_i) v(t_i)$ (cf. (2.5)), so that the computation of the α_k, β_k from

(2.40) and (2.41) is straightforward. In the continuous case, the computation of the inner product requires integration, which complicates matters. Fortunately, for many important special measures $\mathrm{d}\lambda(t) = w(t)\mathrm{d}t$, the recursion coefficients are explicitly known (cf. Chap. 3, Table 3.1). In these cases, it is again straightforward to generate the orthogonal polynomials by (2.38).

The special case of *symmetry* (i.e., $\mathrm{d}\lambda(t) = w(t)\mathrm{d}t$ with $w(-t) = w(t)$ and $\mathrm{supp}(\mathrm{d}\lambda)$ symmetric with respect to the origin) deserves special mention. In this case, defining $p_k(t) = (-1)^k \pi_k(-t)$, one obtains by a simple change of variables that $(p_k, p_\ell) = (-1)^{k+\ell}(\pi_k, \pi_\ell) = 0$ if $k \neq \ell$. Since p_k is monic, it follows by uniqueness that $p_k(t) \equiv \pi_k(t)$; that is,

$$(-1)^k \pi_k(-t) \equiv \pi_k(t) \quad (\mathrm{d}\lambda \text{ symmetric}). \tag{2.43}$$

Thus, if k is even, then π_k is an even polynomial, that is, a polynomial in t^2. Likewise, when k is odd, π_k contains only odd powers of t. As a consequence,

$$\alpha_k = 0 \quad \text{for all } k \geq 0 \quad (\mathrm{d}\lambda \text{ symmetric}), \tag{2.44}$$

which also follows from (2.40), since the numerator on the right-hand side of this equation is an integral of an odd function over a symmetric set of points.

Example: Legendre[3] polynomials.

We may introduce the monic Legendre polynomials by

$$\pi_k(t) = (-1)^k \frac{k!}{(2k)!} \frac{\mathrm{d}^k}{\mathrm{d}t^k}(1-t^2)^k, \quad k = 0, 1, 2, \ldots, \tag{2.45}$$

which is known as the *Rodrigues formula*.

We first verify orthogonality on the interval $[-1, 1]$ relative to the measure $\mathrm{d}\lambda(t) = \mathrm{d}t$. For any ℓ with $0 \leq \ell < k$, repeated integration by parts gives

$$\begin{aligned}\int_{-1}^{1} \frac{\mathrm{d}^k}{\mathrm{d}t^k}(1-t^2)^k \cdot t^\ell \mathrm{d}t &= \sum_{m=0}^{\ell} (-1)^m \ell(\ell-1)\cdots(\ell-m+1)t^{\ell-m} \\ &\quad \times \left.\frac{\mathrm{d}^{k-m-1}}{\mathrm{d}t^{k-m-1}}(1-t^2)^k\right|_{-1}^{1} = 0,\end{aligned}$$

the last equation since $0 \leq k-m-1 < k$. Thus,

$$(\pi_k, p) = 0 \quad \text{for every } p \in \mathbb{P}_{k-1},$$

[3] Adrien Marie Legendre (1752–1833) was a French mathematician active in Paris, best known not only for his treatise on elliptic integrals but also famous for his work in number theory and geometry. He is considered as the originator (in 1805) of the method of least squares, although Gauss had already used it in 1794, but published it only in 1809.

proving orthogonality. Writing (by symmetry)

$$\pi_k(t) = t^k + \mu_k t^{k-2} + \cdots, \quad k \geq 2,$$

and noting (again by symmetry) that the recurrence relation has the form

$$\pi_{k+1}(t) = t\pi_k(t) - \beta_k \pi_{k-1}(t),$$

we obtain

$$\beta_k = \frac{t\pi_k(t) - \pi_{k+1}(t)}{\pi_{k-1}(t)},$$

which is valid for all t. In particular, as $t \to \infty$,

$$\beta_k = \lim_{t\to\infty} \frac{t\pi_k(t) - \pi_{k+1}(t)}{\pi_{k-1}(t)} = \lim_{t\to\infty} \frac{(\mu_k - \mu_{k+1})t^{k-1} + \cdots}{t^{k-1} + \cdots} = \mu_k - \mu_{k+1}.$$

(If $k = 1$, set $\mu_1 = 0$.) From Rodrigues's formula, however, we find

$$\begin{aligned}\pi_k(t) = \frac{k!}{(2k)!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}(t^{2k} - kt^{2k-2} + \cdots) = \frac{k!}{(2k)!}\big(&2k(2k-1)\cdots(k+1)t^k \\ &-k\cdot(2k-2)(2k-3)\cdots(k-1)t^{k-2} + \cdots\big) \\ = t^k - \frac{k(k-1)}{2(2k-1)}t^{k-2} + \cdots,&\end{aligned}$$

so that

$$\mu_k = -\frac{k(k-1)}{2(2k-1)}, \quad k \geq 2.$$

Therefore,

$$\beta_k = \mu_k - \mu_{k+1} = -\frac{k(k-1)}{2(2k-1)} + \frac{(k+1)k}{2(2k+1)} = \frac{k}{2}\,\frac{2k}{(2k+1)(2k-1)};$$

that is, since $\mu_1 = 0$,

$$\beta_k = \frac{1}{4 - k^{-2}}, \quad k \geq 1. \tag{2.46}$$

We conclude with two remarks concerning discrete measures $\mathrm{d}\lambda$ with $\operatorname{supp}\mathrm{d}\lambda = \{t_1, t_2, \ldots, t_N\}$. As before, the L_2 errors decrease monotonically, but the last one is now zero, since there is a polynomial of degree $\leq N-1$ that interpolates f at the N points $t_1, t_2, \ldots, t_N$ (cf. Sect. 2.1.2). Thus,

$$\|f - \hat{\varphi}_0\| \geq \|f - \hat{\varphi}_1\| \geq \cdots \geq \|f - \hat{\varphi}_{N-1}\| = 0, \tag{2.47}$$

where $\hat{\varphi}_n$ is the L_2 approximant of degree $\leq n$,

$$\hat{\varphi}_n(t) = \sum_{j=0}^{n} \hat{c}_j \pi_j(t; d\lambda), \quad \hat{c}_j = \frac{(\pi_j, f)}{(\pi_j, \pi_j)}. \tag{2.48}$$

We see that the polynomial $\hat{\varphi}_{N-1}$ solves the interpolation problem for $\mathbb{P}_{N-1}$. Using (2.48) with $n = N - 1$ to obtain the interpolation polynomial, however, is a roundabout way of solving the interpolation problem. We learn of more direct ways in the next section.

2.2 Polynomial Interpolation

We now wish to approximate functions by matching their values at given points. Using polynomials as approximants gives rise to the following problem: given $n+1$ distinct points $x_0, x_1, \dots, x_n$ and values $f_i = f(x_i)$ of some function f at these points, find a polynomial $p \in \mathbb{P}_n$ such that

$$p(x_i) = f_i, \quad i = 0, 1, 2, \dots, n.$$

Since we have to satisfy $n + 1$ conditions, and have at our disposal $n + 1$ degrees of freedom – the coefficients of p – we expect the problem to have a unique solution. Other questions of interest, in addition to existence and uniqueness, are different ways of representing and computing the polynomial p, what can be said about the error $e(x) = f(x) - p(x)$ when $x \neq x_i$, $i = 0, 1, \dots, n$, and the quality of approximation $f(x) \approx p(x)$ when the number of points, and hence the degree of p, is allowed to increase indefinitely. Although these questions are not of the utmost interest in themselves, the results discussed here are widely used in the development of approximate methods for more important practical tasks such as solving initial and boundary value problems for ordinary and partial differential equations. It is in view of these and other applications that we study polynomial interpolation.

The simplest example is *linear interpolation*, that is, the case $n = 1$. Here, it is obvious from Fig. 2.3 that the interpolation problem has a unique solution. It is also clear that the error $e(x)$ can be as large as one likes (or dislikes) if nothing is known about f other than its two values at x_0 and x_1.

One way of writing down the linear interpolant p is as a weighted average of f_0 and f_1 (already taught in high school),

$$p(x) = \frac{x - x_1}{x_0 - x_1} f_0 + \frac{x - x_0}{x_1 - x_0} f_1.$$

This is the way Lagrange expressed p in the general case (cf. Sect. 2.1.2). However, we can write p also in Taylor's form, noting that its derivative at x_0 is equal to the "difference quotient,"

$$p(x) = f_0 + \frac{f_1 - f_0}{x_1 - x_0}(x - x_0).$$

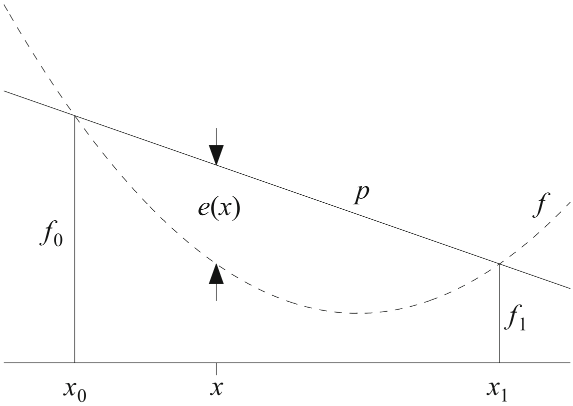


Fig. 2.3 Linear interpolation

This indeed is a prototype of Newton's form of the interpolation polynomial (cf. Sect. 2.2.6).

Interpolating to function values is referred to as *Lagrange interpolation*. More generally, we may wish to interpolate to function and consecutive derivative values of some function. This is called *Hermite interpolation*. It turns out that the latter can be solved as a limit case of the former (cf. Sect. 2.2.7).

2.2.1 Lagrange Interpolation Formula: Interpolation Operator

We prove the existence of the interpolation polynomial by simply writing it down. It is clear, indeed, that

$$\ell_i(x) = \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j}, \quad i = 0, 1, \ldots, n, \tag{2.49}$$

is a polynomial of degree n that interpolates to 1 at $x = x_i$ and to 0 at all the other points. Multiplying it by f_i produces the correct value at x_i, and then adding up the resulting polynomials,

$$p(x) = \sum_{i=0}^{n} f_i \ell_i(x),$$

produces a polynomial, still of degree $\leq n$, that has the desired interpolation properties. To prove this formally, note that

$$\ell_i(x_k) = \delta_{ik} = \begin{cases} 1 & \text{if} \quad i = k, \\ 0 & \text{if} \quad i \neq k, \end{cases} \quad i, k = 0, 1, \dots, n. \tag{2.50}$$

Therefore,

$$p(x_k) = \sum_{i=0}^{n} f_i \ell_i(x_k) = \sum_{i=0}^{n} f_i \delta_{ik} = f_k, \quad k = 0, 1, \dots, n.$$

This establishes the *existence* of the interpolation polynomial. To prove *uniqueness*, assume that there are two polynomials of degree $\leq n$, say, p and p^*, both interpolating to f at x_i, $i = 0, 1, \dots, n$. Then

$$d(x) = p(x) - p^*(x)$$

is a polynomial of degree $\leq n$ that satisfies

$$d(x_i) = f_i - f_i = 0, \quad i = 0, 1, \dots, n.$$

In other words, d has $n + 1$ distinct zeros x_i. There is only *one* polynomial in $\mathbb{P}_n$ with that many zeros, namely, $d(x) \equiv 0$. Therefore, $p^*(x) \equiv p(x)$.

We denote the unique polynomial $p \in \mathbb{P}_n$ interpolating f at the (distinct) points $x_0, x_1, \dots, x_n$ by

$$p_n(f; x_0, x_1, \dots, x_n; x) = p_n(f; x), \tag{2.51}$$

where we use the long form on the left-hand side if we want to place in evidence the points at which interpolation takes place, and the short form on the right-hand side if the choice of these points is clear from the context. We thus have what is called the *Lagrange*[4] *interpolation formula*

$$p_n(f; x) = \sum_{i=0}^{n} f(x_i)\ell_i(x), \tag{2.52}$$

with the $\ell_i(x)$ – the *elementary Lagrange interpolation polynomials* – defined in (2.49).

---

[4]Joseph Louis Lagrange (1736–1813), born in Turin, became, through correspondence with Euler, his protégé. In 1766 he indeed succeeded Euler in Berlin. He returned to Paris in 1787. Clairaut wrote of the young Lagrange: "... a young man, no less remarkable for his talents than for his modesty; his temperament is mild and melancholic; he knows no other pleasure than study." Lagrange made fundamental contributions to the calculus of variations and to number theory, and worked also on many problems in analysis. He is widely known for his representation of the remainder term in Taylor's formula. The interpolation formula appeared in 1794. His *Mécanique Analytique*, published in 1788, made him one of the founders of analytic mechanics.

It is useful to look at Lagrange interpolation in terms of a (linear) operator P_n from (say) the space of continuous functions to the space of polynomials $\mathbb{P}_n$,

$$P_n : \quad C[a,b] \to \mathbb{P}_n, \quad p(\cdot) = p_n(f; \cdot). \tag{2.53}$$

The interval $[a,b]$ here is any interval containing all points x_i, $i = 0, 1, \dots, n$. The operator P_n has the following properties:

1. $P_n(\alpha f) = \alpha P_n f, \ \ \alpha \in \mathbb{R}$ (homogeneity);
2. $P_n(f + g) = P_n f + P_n g$ (additivity).

Combining 1 and 2 shows that P_n is a linear operator,

$$P_n(\alpha f + \beta g) = \alpha P_n f + \beta P_n g, \quad \alpha, \beta \in \mathbb{R}.$$

3. $P_n f = f$ for all $f \in \mathbb{P}_n$.

The last property – an immediate consequence of uniqueness of the interpolation polynomial – says that P_n leaves polynomials of degree $\leq n$ unchanged, and hence is a *projection* operator.

A norm of the linear operator P_n can be defined (similarly as for matrices, cf. Chap. 1, (1.30)) by

$$\|P_n\| = \max_{f \in C[a,b]} \frac{\|P_n f\|}{\|f\|}, \tag{2.54}$$

where on the right-hand side one takes any convenient norm for functions. Taking the L_∞ norm (cf. Table 2.1), one obtains from Lagrange's formula (2.52)

$$\begin{aligned} \|p_n(f; \cdot)\|_\infty &= \max_{a \leq x \leq b} \left| \sum_{i=0}^{n} f(x_i)\ell_i(x) \right| \\ &\leq \|f\|_\infty \max_{a \leq x \leq b} \sum_{i=0}^{n} |\ell_i(x)|. \end{aligned} \tag{2.55}$$

Indeed, equality holds for some continuous function f; cf. Ex. 30. Therefore,

$$\|P_n\|_\infty = \Lambda_n, \tag{2.56}$$

where

$$\Lambda_n = \|\lambda_n\|_\infty, \quad \lambda_n(x) = \sum_{i=0}^{n} |\ell_i(x)|. \tag{2.57}$$

The function $\lambda_n(x)$ and its maximum Λ_n are called, respectively, the *Lebesgue*[5] *function* and *Lebesgue constant* for Lagrange interpolation. They provide a first estimate for the interpolation error: let $\mathcal{E}_n(f)$ be the *best* (uniform) *approximation* of f on $[a, b]$ by polynomials of degree $\le n$,

$$\mathcal{E}_n(f) = \min_{p \in \mathbb{P}_n} \|f - p\|_\infty = \|f - \hat{p}_n\|_\infty, \tag{2.58}$$

where $\hat{p}_n$ is the nth-degree polynomial of best uniform approximation to f. Then, using the basic properties 1–3 of P_n, in particular, the projection property 3, and (2.55) and (2.57), one finds

$$\begin{aligned} \|f - p_n(f;\cdot)\|_\infty &= \|f - \hat{p}_n - p_n(f - \hat{p}_n;\cdot)\|_\infty \\ &\le \|f - \hat{p}_n\|_\infty + \Lambda_n \|f - \hat{p}_n\|_\infty; \end{aligned}$$

that is,

$$\|f - p_n(f;\cdot)\|_\infty \le (1 + \Lambda_n)\mathcal{E}_n(f). \tag{2.59}$$

Thus, the better f can be approximated by polynomials of degree $\le n$, the smaller the interpolation error. Unfortunately, Λ_n is not uniformly bounded: no matter how one chooses the nodes $x_i = x_i^{(n)}$, $i = 0, 1, \ldots, n$, one can show that always $\Lambda_n > O(\log n)$ as $n \to \infty$. It is not possible, therefore, to conclude from Weierstrass's approximation theorem (i.e., from $\mathcal{E}_n(f) \to 0$, $n \to \infty$) that Lagrange interpolation converges uniformly on $[a, b]$ for any continuous function, not even for judiciously selected nodes; indeed, one knows that it does not.

2.2.2 Interpolation Error

As noted earlier, we need to make some assumptions about the function f in order to be able to estimate the error of interpolation, $f(x) - p_n(f; x)$, for any $x \ne x_i$ in $[a, b]$. In (2.59) we made an assumption in terms of how well f can be approximated on $[a, b]$ by polynomials of degree $\le n$. Now we make an assumption on the magnitude of some appropriate derivative of f.

[5] Henri Leon Lebesgue (1875–1941) was a French mathematician best known for his work on the theory of real functions, notably the concepts of measure and integral that now bear his name. These became fundamental in many areas of mathematics such as functional analysis, Fourier analysis, and probability theory. He has also made interesting contributions to the calculus of variations, the theory of dimension, and set theory.

It is not difficult to guess how the formula for the error should look: since the error is zero at each x_i, $i = 0, 1, \ldots, n$, we ought to see a factor of the form $(x - x_0)(x - x_1) \cdots (x - x_n)$. On the other hand, by the projection property 3 in Sect. 2.2.1, the error is also zero (even identically so) if $f \in \mathbb{P}_n$, which suggests another factor – the $(n+1)$st derivative of f. But evaluated where? Certainly not at x, since f would then have to satisfy a differential equation. So let us say that $f^{(n+1)}$ is evaluated at some point $\xi = \xi(x)$, which is unknown but must be expected to depend on x. Now if we test the formula so far conjectured on the simplest nontrivial polynomial, $f(x) = x^{n+1}$, we discover that a factor $1/(n+1)!$ is missing. So, our final (educated) guess is the formula

$$f(x) - p_n(f;x) = \frac{f^{(n+1)}(\xi(x))}{(n+1)!} \prod_{i=0}^{n} (x - x_i), \quad x \in [a,b]. \tag{2.60}$$

Here $\xi(x)$ is some number in the open interval (a,b), but otherwise unspecified,

$$a < \xi(x) < b. \tag{2.61}$$

The statement (2.60) and (2.61) is, in fact, correct if we assume that $f \in C^{n+1}[a,b]$. An elegant proof of it, due to Cauchy,[6] goes as follows. We can assume $x \neq x_i$ for $i = 0, 1, \ldots, n$, since otherwise (2.60) would be trivially true for any $\xi(x)$. So, fix $x \in [a,b]$ in this manner, and define a function F of the new variable t as follows:

$$F(t) = f(t) - p_n(f;t) - \frac{f(x) - p_n(f;x)}{\prod_{i=0}^{n}(x - x_i)} \prod_{i=0}^{n} (t - x_i). \tag{2.62}$$

Clearly, $F \in C^{n+1}[a,b]$. Furthermore,

$$F(x_i) = 0, \; i = 0, 1, \ldots, n; \quad F(x) = 0.$$

Thus, F has $n+2$ distinct zeros in $[a,b]$. Applying repeatedly Rolle's Theorem, we conclude that

[6]Augustin Louis Cauchy (1789–1857), active in Paris, is truly the father of modern analysis. He provided a firm foundation for analysis by basing it on a rigorous concept of limit. He is also the creator of complex analysis, of which "Cauchy's formula" (cf. (2.70)) is a centerpiece. In addition, Cauchy's name is attached to pioneering contributions to the theory of ordinary and partial differential equations, in particular, regarding questions of existence and uniqueness. As with many great mathematicians of the eighteenth and nineteenth centuries, his work also encompasses geometry, algebra, number theory, and mechanics, as well as theoretical physics.

$$
\begin{array}{lll}
F' & \text{has at least } n+1 & \text{distinct zeros in } (a,b) \\
F'' & \text{has at least } n & \text{distinct zeros in } (a,b) \\
F''' & \text{has at least } n-1 & \text{distinct zeros in } (a,b) \\
\cdots & \cdots\cdots\cdots\ \cdots & \cdots\ \cdots\ \cdots\cdots\ \cdots \\
F^{(n+1)} & \text{has at least } 1 & \text{zero in } (a,b)
\end{array}
$$

since $F^{(n+1)}$ is still continuous on $[a,b]$. Denote by $\xi(x)$ a zero of $F^{(n+1)}$ whose existence we just established. It certainly satisfies (2.61) and, of course, will depend on x. Now differentiating F in (2.62) $n+1$ times with respect to t, and then setting $t = \xi(x)$, we get

$$0 = f^{(n+1)}(\xi(x)) - \frac{f(x) - p_n(f;x)}{\prod_{i=0}^{n}(x - x_i)} \cdot (n+1)!,$$

which, when solved for $f(x) - p_n(f;x)$, gives precisely (2.60). Actually, what we have shown is that $\xi(x)$ is contained in the span of $x_0, x_1, \ldots, x_n, x$, that is, in the interior of the smallest closed interval containing $x_0, x_1, \ldots, x_n$ and x.

Examples. 1. Linear interpolation ($n = 1$). Assume that $x_0 \le x \le x_1$; that is, $[a,b] = [x_0, x_1]$, and let $h = x_1 - x_0$. Then by (2.60) and (2.61),

$$f(x) - p_1(f;x) = (x - x_0)(x - x_1)\frac{f''(\xi)}{2}, \quad x_0 < \xi < x_1,$$

and an easy computation gives

$$\|f - p_1(f;\cdot)\|_\infty \le \frac{M_2}{8}h^2, \quad M_2 = \|f''\|_\infty. \tag{2.63}$$

Here the ∞-norm refers to the interval $[x_0, x_1]$. Thus, on small intervals of length h, the error for linear interpolation is $O(h^2)$.

2. Quadratic interpolation ($n = 2$) on *equally spaced* points x_0, $x_1 = x_0 + h$, $x_2 = x_0 + 2h$. We now have, for $x \in [x_0, x_2]$,

$$f(x) - p_2(f;x) = (x - x_0)(x - x_1)(x - x_2)\frac{f'''(\xi)}{6}, \quad x_0 < \xi < x_2,$$

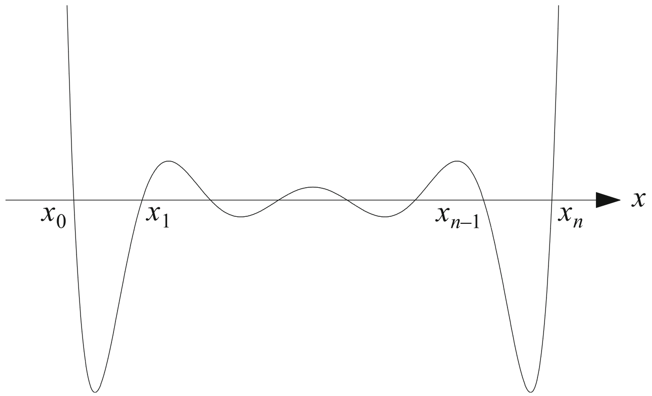


Fig. 2.4 Interpolation error for eight equally spaced points

and (cf. Ex. 43(a))

$$\|f - p_2(f;\cdot)\|_\infty \le \frac{M_3}{9\sqrt{3}}h^3, \quad M_3 = \|f'''\|_\infty,$$

giving an error of $O(h^3)$.

3. nth-degree interpolation on *equally spaced* points $x_i = x_0 + ih$, $i = 0, 1, \dots, n$. When h is small, and $x_0 \le x \le x_n$, then $\xi(x)$ in (2.60) is constrained to a relatively small interval and $f^{(n+1)}(\xi(x))$ cannot vary a great deal. The behavior of the error, therefore, is mainly determined by the product $\prod_{i=0}^{n}(x - x_i)$, the graph of which, for $n = 7$, is shown in Fig. 2.4. We clearly have symmetry with respect to the midpoint $(x_0 + x_n)/2$. It can also be shown that the relative extrema decrease monotonically in modulus as one moves from the endpoints to the center (cf. Ex. 29(c)).

   It is evident that the oscillations become more violent as n increases. In particular, the curve is extremely steep at the endpoints, and takes off to ∞ rapidly as x moves away from the interval $[x_0, x_n]$. Although it is true that the curve representing the interpolation error is scaled by a factor of $O(h^{n+1})$, it is also clear that one ought to interpolate near the center zone of the interval $[x_0, x_n]$, if at all possible, and should avoid interpolation near the end zones, or even *extra*polation outside the interval. The highly oscillatory nature of the error curve, when n is large, also casts some legitimate doubts about convergence of the interpolation process as $n \to \infty$. This is studied in the next section.

2.2.3 Convergence

We first must define what we mean by "convergence." We assume that we are given a triangular array of interpolation nodes $x_i = x_i^{(n)}$, exactly $n + 1$ distinct nodes for each $n = 0, 1, 2, \ldots$:

$$
\begin{array}{llllll}
x_0^{(0)} & & & & & \\
x_0^{(1)} & x_1^{(1)} & & & & \\
x_0^{(2)} & x_1^{(2)} & x_2^{(2)} & & & \\
\cdot\;\cdot\;\cdot & \cdot\;\cdot\;\cdot & \cdot\;\cdot\;\cdot & \cdot\;\cdot\;\cdot & & \\
x_0^{(n)} & x_1^{(n)} & x_2^{(n)} & \cdots & x_n^{(n)} & \\
\cdot\;\cdot\;\cdot & \cdot\;\cdot\;\cdot & \cdot\;\cdot\;\cdot & \cdot\;\cdot\;\cdot & \cdot\;\cdot\;\cdot & \cdot\;\cdot\;\cdot
\end{array}
\tag{2.64}
$$

We further assume that all nodes $x_i^{(n)}$ are contained in some finite interval $[a, b]$. Then, for each n, we define

$$p_n(x) = p_n(f; x_0^{(n)}, x_1^{(n)}, \ldots, x_n^{(n)}; x), \quad x \in [a, b]. \tag{2.65}$$

We say that Lagrange interpolation based on the triangular array of nodes (2.64) *converges* if

$$p_n(x) \to f(x) \text{ as } n \to \infty, \tag{2.66}$$

uniformly for $x \in [a, b]$.

Convergence clearly depends on the behavior of the kth derivative $f^{(k)}$ of f as $k \to \infty$. We assume that $f \in C^\infty[a, b]$, and that

$$|f^{(k)}(x)| \le M_k \text{ for } a \le x \le b, \quad k = 0, 1, 2, \ldots. \tag{2.67}$$

Since $|x - x_i^{(n)}| \le b - a$ whenever $x \in [a, b]$ and $x_i^{(n)} \in [a, b]$, we have

$$|(x - x_0^{(n)})(x - x_1^{(n)}) \cdots (x - x_n^{(n)})| \le (b - a)^{n+1}, \tag{2.68}$$

so that by (2.60)

$$|f(x) - p_n(x)| \le (b - a)^{n+1} \frac{M_{n+1}}{(n + 1)!}, \quad x \in [a, b].$$

We therefore have convergence if

$$\lim_{k \to \infty} \frac{(b - a)^k}{k!} M_k = 0. \tag{2.69}$$

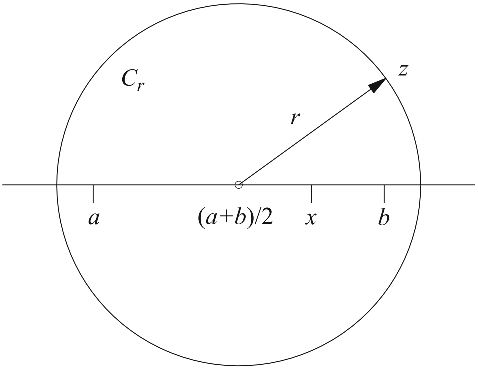


Fig. 2.5 The circular disk C_r

We now show that (2.69) is true if f is analytic in a sufficiently large region in the complex plane containing the interval $[a,b]$. Specifically, let C_r be the circular (closed) disk with center at the midpoint of $[a,b]$ and radius r, and assume, for the time being, that $r > \frac{1}{2}(b-a)$, so that $[a,b] \subset C_r$. Assume f analytic in C_r. Then we can estimate the derivative in (2.67) by Cauchy's Formula,

$$f^{(k)}(x) = \frac{k!}{2\pi \mathrm{i}} \oint_{\partial C_r} \frac{f(z)}{(z-x)^{k+1}}\,\mathrm{d}z, \quad x \in [a,b]. \tag{2.70}$$

Noting that $|z-x| \geq r - \frac{1}{2}(b-a)$ (cf. Fig. 2.5), we obtain

$$|f^{(k)}(x)| \leq \frac{k!}{2\pi} \frac{\max\limits_{z \in \partial C_r} |f(z)|}{[r - \frac{1}{2}(b-a)]^{k+1}} \cdot 2\pi r.$$

Therefore, we can take for M_k in (2.67)

$$M_k = \frac{r}{r - \frac{1}{2}(b-a)} \max_{z \in \partial C_r} |f(z)| \cdot \frac{k!}{[r - \frac{1}{2}(b-a)]^k}, \tag{2.71}$$

and (2.69) holds if

$$\left(\frac{b-a}{r - \frac{1}{2}(b-a)} \right)^k \to 0 \text{ as } k \to \infty,$$

that is, if $b - a < r - \frac{1}{2}(b-a)$, or, equivalently,

$$r > \tfrac{3}{2}(b-a). \tag{2.72}$$

We have shown that *Lagrange interpolation converges* (uniformly on $[a,b]$) *for an arbitrary triangular set of nodes* (2.64) (all contained in $[a,b]$) *if f is analytic in the circular disk C_r centered at $(a+b)/2$ and having radius r sufficiently large so that* (2.72) *holds*.

Since our derivation of this result used rather crude estimates (see, in particular, (2.68)), the required domain of analyticity for f that we found is certainly not sharp. Using more refined methods, one can prove the following. Let $\mathrm{d}\mu(t)$ be the "limit distribution" of the interpolation nodes, that is, let

$$\int_a^x \mathrm{d}\mu(t), \quad a < x \le b,$$

be the ratio of the number of nodes $x_i^{(n)}$ in $[a,x]$ to the total number, $n+1$, of nodes, asymptotically as $n \to \infty$. (When the nodes are uniformly distributed over the interval $[a,b]$, then $\mathrm{d}\mu(t) = \mathrm{d}t/(b-a)$.) A curve of *constant logarithmic potential* is the locus of all complex $z \in \mathbb{C}$ such that

$$u(z) = \gamma, \;\; u(z) = \int_a^b \ln \frac{1}{|z-t|}\, \mathrm{d}\mu(t),$$

where γ is a constant. For large negative γ, these curves look like circles with large radii and center at $(a+b)/2$. As γ increases, the curves "shrink" toward the interval $[a,b]$. Let

$$\Gamma = \sup \gamma,$$

where the supremum is taken over all curves $u(z) = \gamma$ containing $[a,b]$ in their interior. The important domain (replacing C_r) is then the domain

$$C_\Gamma = \{z \in \mathbb{C} : \;\; u(z) \ge \Gamma\}, \tag{2.73}$$

in the sense that *if f is analytic in any domain C containing C_Γ in its interior* (no matter how closely C covers C_Γ), *then*

$$|f(z) - p_n(f;z)| \to 0 \;\; as \;\; n \to \infty \tag{2.74}$$

uniformly for $z \in C_\Gamma$.

Examples. 1. Equally distributed nodes: $\mathrm{d}\mu(t) = \mathrm{d}t/(b-a)$, $a \le t \le b$. In this case, C_Γ is a lens-shaped domain with tips at a and b, as shown in Fig. 2.6. Thus, we have uniform convergence in C_Γ (not just on $[a,b]$, as before) provided f is analytic in a region slightly larger than C_Γ.

2. Arc sine distribution on [–1,1]: $\mathrm{d}\mu(t) = \dfrac{1}{\pi}\dfrac{\mathrm{d}t}{\sqrt{1-t^2}}$. Here the nodes are more densely distributed near the endpoints of the interval [–1,1]. It turns out that in this case C_Γ = [–1,1], so that Lagrange interpolation converges uniformly on [–1,1] if f is "analytic on [1,1]," that is, analytic in any region, no matter how thin, that contains the interval [–1,1] in its interior.

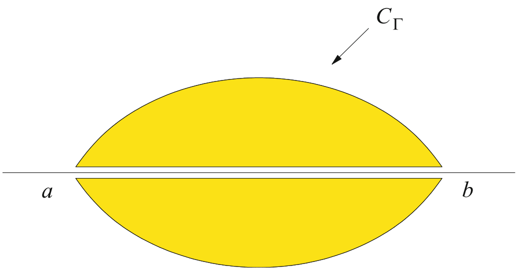


Fig. 2.6 The domain C_Γ for uniformly distributed nodes

3. Runge's[7] example:

$$f(x) = \frac{1}{1+x^2}, \quad -5 \le x \le 5,$$
$$x_k^{(n)} = -5 + k\frac{10}{n}, \quad k = 0, 1, 2, \ldots, n. \tag{2.75}$$

Here the nodes are equally spaced, hence asymptotically equally distributed. Note that $f(z)$ has poles at $z = \pm\,\mathrm{i}$. These poles lie definitely inside the region C_Γ in Fig. 2.6 for the interval [–5,5], so that f is *not* analytic in C_Γ. For this reason, we can no longer expect convergence on the whole interval [–5,5]. It has been shown, indeed, that

$$\lim_{n\to\infty} |f(x) - p_n(f;x)| = \begin{cases} 0 & \text{if } |x| < 3.633\ldots, \\ \infty & \text{if } |x| > 3.633\ldots. \end{cases} \tag{2.76}$$

We have convergence in the central zone of the interval [–5,5], but divergence in the lateral zones. With Fig. 2.4 kept in mind, this is perhaps not all that surprising (cf. MA 7(b)).

4. Bernstein's[8] example:

$$f(x) = |x|, \quad -1 \le x \le 1,$$
$$x_k^{(n)} = -1 + \frac{2k}{n}, \quad k = 0, 1, 2, \ldots, n. \tag{2.77}$$

---

[7]Carl David Tolme Runge (1856–1927) was active in the famous Göttingen school of mathematics and is one of the early pioneers of numerical mathematics. He is best known for the Runge–Kutta formula in ordinary differential equations (cf. Chap. 5, Sect. 5.6.5), for which he provided the basic idea. He made also notable contributions to approximation theory in the complex plane.

[8]Sergei Natanovič Bernštein (1880–1968) made major contributions to polynomial approximation, continuing in the tradition of his countryman Chebyshev. He is also known for his work on partial differential equations and probability theory.

Here analyticity of f is completely gone, f being not even differentiable at $x = 0$. Accordingly, one finds that

$$\lim_{n\to\infty} |f(x) - p_n(f;x)| = \infty \text{ for every } x \in [-1,1], \\ \text{except } x = -1,\ x = 0, \text{ and } x = 1. \tag{2.78}$$

The fact that $x = \pm 1$ are exceptional points is trivial, since they are interpolation nodes, where the error is zero. The same is true for $x = 0$ when n is even, but not if n is odd.

The failure of convergence in the last two examples can only in part be blamed on insufficient regularity of f. Another culprit is the equidistribution of the nodes. There are indeed better distributions, for example, the arc sine distribution of Example 2. An instance of the latter is discussed in the next section.

We add one more example, which involves *complex* nodes, and for which the preceding theory, therefore, no longer applies. We prove convergence directly.

5. Interpolation at the roots of unity (Fejér[9]): $z_k = \exp(k2\pi \mathrm{i}/n)$, $k = 1, 2, \ldots, n$. We show that

$$p_{n-1}(f;z) \to f(z),\ \ n \to \infty, \ \text{ for any } |z| < 1, \tag{2.79}$$

uniformly in any disk $|z| \le \rho < 1$, provided f is analytic in $|z| < 1$ and continuous on $|z| \le 1$.

We have

$$\omega_n(z) := \prod_{k=1}^{n} (z - z_k) = z^n - 1,\ \ \omega_n'(z_k) = n z_k^{n-1} = \frac{n}{z_k},$$

so that the elementary Lagrange polynomials are

$$\ell_k(z) = \frac{\omega_n(z)}{\omega_n'(z_k)(z - z_k)} = \frac{z^n - 1}{\frac{n}{z_k}(z - z_k)} \\ = \frac{z_k}{n}\frac{1}{z_k - z} + z^n \frac{z_k}{(z - z_k)n}.$$

[9] Leopold Fejér (1880–1959) was a leading Hungarian mathematician of the twentieth century. Interestingly, Fejér had great difficulties in mathematics at the elementary and lower secondary school level, and even required private tutoring. It was an inspiring teacher in the upper-level secondary school who awoke Fejér's interest and passion for mathematics. He went on to discover – still a university student – an important result on the summability of Fourier series, which made him famous overnight. He continued to make further contributions to the theory of Fourier series, but also occupied himself with problems of approximation and interpolation in the real as well as complex domain. He in turn was an inspiring teacher to the next generation of Hungarian mathematicians.

Therefore,

$$p_{n-1}(f;z) = \sum_{k=1}^{n} \frac{f(z_k)}{z_k - z}\frac{z_k}{n} + z^n \sum_{k=1}^{n} \frac{f(z_k)}{z - z_k}\frac{z_k}{n}. \tag{2.80}$$

We interpret the first sum as a Riemann sum of an integral extended over the unit circle:

$$\sum_{k=1}^{n} \frac{f(z_k)}{z_k - z}\frac{z_k}{n} = \frac{1}{2\pi \mathrm{i}} \sum_{k=1}^{n} \frac{f(\mathrm{e}^{\mathrm{i}k2\pi/n})}{\mathrm{e}^{\mathrm{i}k2\pi/n} - z}\, \mathrm{i}\mathrm{e}^{\mathrm{i}k2\pi/n} \cdot \frac{2\pi}{n}$$

$$\to \frac{1}{2\pi \mathrm{i}} \int_0^{2\pi} \frac{f(\mathrm{e}^{\mathrm{i}\theta})}{\mathrm{e}^{\mathrm{i}\theta} - z}\, \mathrm{i}\mathrm{e}^{\mathrm{i}\theta}\mathrm{d}\theta = \frac{1}{2\pi \mathrm{i}} \oint_{|\zeta|=1} \frac{f(\zeta)\mathrm{d}\zeta}{\zeta - z} \quad \text{as } n \to \infty.$$

The last expression, by Cauchy's Formula, however, is precisely $f(z)$. The second term in (2.80), being just $-z^n$ times the first, converges to zero, uniformly in $|z| \le \rho < 1$.

2.2.4 Chebyshev Polynomials and Nodes

The choice of nodes, as we saw in the previous section, distinctly influences the convergence character of the interpolation process. We now discuss a choice of points – the *Chebyshev points* – which leads to very favorable convergence properties. These points are useful, not only for interpolation, but also for other purposes (integration, collocation, etc.). We consider them on the canonical interval [–1,1], but they can be defined on any finite interval $[a, b]$ by means of a linear transformation of variables that maps [–1,1] onto $[a, b]$.

We begin with developing the Chebyshev polynomials. They arise from the fact that the cosine of a multiple argument is a polynomial in the cosine of the simple argument; more precisely,

$$\cos n\theta = T_n(\cos\theta), \quad T_n \in \mathbb{P}_n. \tag{2.81}$$

This is a consequence of the well-known trigonometric identity

$$\cos(k+1)\theta + \cos(k-1)\theta = 2\cos\theta\cos k\theta,$$

which, when solved for the first term, gives

$$\cos(k+1)\theta = 2\cos\theta\cos k\theta - \cos(k-1)\theta. \tag{2.82}$$

Therefore, if $\cos m\theta$ is a polynomial of degree m in $\cos\theta$ for all $m \le k$, then the same is true for $m = k+1$. Mathematical induction then proves (2.81). At the same time, it follows from (2.81) and (2.82), if we set $\cos\theta = x$, that

$$\begin{aligned} T_{k+1}(x) &= 2xT_k(x) - T_{k-1}(x), \quad k = 1, 2, 3, \ldots, \\ T_0(x) &= 1, \quad T_1(x) = x. \end{aligned} \tag{2.83}$$

The polynomials T_m so defined are called the *Chebyshev polynomials* (of the first kind). Thus, for example,

$$T_2(x) = 2x^2 - 1,$$

$$T_3(x) = 4x^3 - 3x,$$

$$T_4(x) = 8x^4 - 8x^2 + 1,$$

and so on.

Clearly, these polynomials are defined not only for x in [–1,1], but also for arbitrary real or complex x. It is only that on the interval [–1,1] they satisfy the identity (2.81) (where θ is real).

It is evident from (2.83) that the leading coefficient of T_n is 2^{n-1} (if $n \geq 1$); the *monic Chebyshev polynomial* of degree n, therefore, is

$$\overset{\circ}{T}_n(x) = \frac{1}{2^{n-1}} T_n(x), \ n \geq 1; \quad \overset{\circ}{T}_0 = T_0. \tag{2.84}$$

The basic identity (2.81) allows us to immediately obtain the zeros $x_k = x_k^{(n)}$ of T_n: indeed, $\cos n\theta = 0$ if $n\theta = (2k-1)\pi/2$, so that

$$x_k^{(n)} = \cos \theta_k^{(n)}, \quad \theta_k^{(n)} = \frac{2k-1}{2n}\pi, \quad k = 1, 2, \ldots, n. \tag{2.85}$$

All zeros of T_n are thus real, distinct, and contained in the open interval (–1,1). They are the projections onto the real line of equally spaced points on the unit circle; cf. Fig. 2.7 for the case $n = 4$.

In terms of the zeros $x_k^{(n)}$ of T_n, we can write the monic polynomial in factored form as

$$\overset{\circ}{T}_n(x) = \prod_{k=1}^{n} \left(x - x_k^{(n)} \right). \tag{2.86}$$

As we let θ increase from 0 to π, hence $x = \cos\theta$ decrease from $+1$ to -1, (2.81) shows that $T_n(x)$ oscillates between +1 and –1, attaining these extreme values at

$$y_k^{(n)} = \cos \eta_k^{(n)}, \quad \eta_k^{(n)} = k\frac{\pi}{n}, \quad k = 0, 1, 2, \ldots, n. \tag{2.87}$$

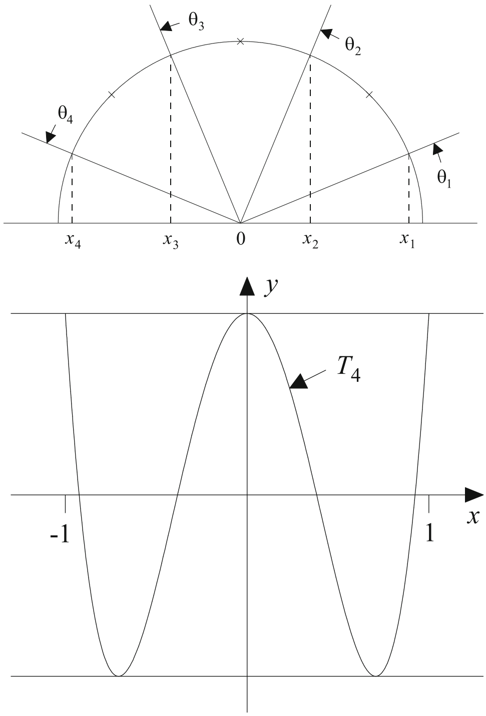


Fig. 2.7 The Chebyshev polynomial $y = T_4(x)$

In summary, then,

$$T_n\left(x_k^{(n)}\right) = 0 \text{ for } x_k^{(n)} = \cos\frac{2k-1}{2n}\pi, \quad k = 1, 2, \ldots, n; \tag{2.88}$$

$$T_n\left(y_k^{(n)}\right) = (-1)^k \text{ for } y_k^{(n)} = \cos\frac{k}{n}\pi, \quad k = 0, 1, 2, \ldots, n. \tag{2.89}$$

Chebyshev polynomials owe their importance and usefulness to the following theorem, due to Chebyshev.[10]

Theorem 2.2.1. *For an arbitrary monic polynomial $\overset{\circ}{p}_n$ of degree n, there holds*

$$\max_{-1\le x\le 1} |\overset{\circ}{p}_n(x)| \ge \max_{-1\le x\le 1} |\overset{\circ}{T}_n(x)| = \frac{1}{2^{n-1}}, \quad n \ge 1, \tag{2.90}$$

where $\overset{\circ}{T}_n$ is the monic Chebyshev polynomial (2.84) *of degree n.*

[10]Pafnuti Levovich Chebyshev (1821–1894) was the most prominent member of the St. Petersburg school of mathematics. He made pioneering contributions to number theory, probability theory, and approximation theory. He is regarded as the founder of constructive function theory, but also worked in mechanics, notably the theory of mechanisms, and in ballistics.

Proof (by contradiction). Assume, contrary to (2.90), that

$$\max_{-1 \le x \le 1} |\overset{\circ}{p}_n(x)| < \frac{1}{2^{n-1}}. \tag{2.91}$$

Then the polynomial $d_n(x) = \overset{\circ}{T}_n(x) - \overset{\circ}{p}_n(x)$ (a polynomial of degree $\le n-1$), satisfies

$$d_n\left(y_0^{(n)}\right) > 0,\ d_n\left(y_1^{(n)}\right) < 0,\ d_n\left(y_2^{(n)}\right) > 0, \dots,\ (-1)^n d_n\left(y_n^{(n)}\right) > 0. \tag{2.92}$$

Thus d_n changes sign at least n times, and hence has at least n distinct real zeros. But having degree $\le n-1$, it must vanish identically, $d_n(x) \equiv 0$. This contradicts (2.92); thus (2.91) cannot be true. □

The result (2.90) can be given the following interesting interpretation: *the best uniform approximation* (on the interval [–1,1]) *to* $f(x) = x^n$ *from polynomials in* $\mathbb{P}_{n-1}$ *is given by* $x^n - \overset{\circ}{T}_n(x)$, that is, by the aggregate of terms of degree $\le n-1$ in $\overset{\circ}{T}_n$ taken with the minus sign. From the theory of uniform polynomial approximation it is known that the best approximant is unique. Therefore, equality in (2.90) can only hold if $\overset{\circ}{p}_n = \overset{\circ}{T}_n$.

What is the significance of Chebyshev polynomials for interpolation? Recall (cf. (2.60)) that the interpolation error (on [–1,1], for a function $f \in C^{n+1}[-1,1]$), is given by

$$f(x) - p_n(f;x) = \frac{f^{(n+1)}(\xi(x))}{(n+1)!} \cdot \prod_{i=0}^{n}(x - x_i), \quad x \in [-1,1]. \tag{2.93}$$

The first factor is essentially independent of the choice of the nodes x_i. It is true that $\xi(x)$ *does* depend on the x_i, but we usually estimate $f^{(n+1)}$ by $\|f^{(n+1)}\|_\infty$, which removes this dependence. On the other hand, the product in the second factor, including its norm

$$\left\| \prod_{i=0}^{n}(\cdot - x_i) \right\|_\infty, \tag{2.94}$$

depends strongly on the x_i. It makes sense, therefore, to try to minimize (2.94) over all $x_i \in [-1,1]$. Since the product in (2.94) is a monic polynomial of degree $n+1$, it follows from Theorem 2.2.1 that *the optimal nodes* $x_i = \hat{x}_i^{(n)}$ in (2.93) *are precisely the zeros of* T_{n+1}; that is,

$$\hat{x}_i^{(n)} = \cos\frac{2i+1}{2n+2}\pi, \quad i = 0, 1, 2, \dots, n. \tag{2.95}$$

For these nodes, we then have (cf. (2.90))

$$\|f(\cdot) - p_n(f;\cdot)\|_\infty \le \frac{\|f^{(n+1)}\|_\infty}{(n+1)!} \cdot \frac{1}{2^n}. \tag{2.96}$$

One ought to compare the last factor in (2.96) with the much cruder bound given in (2.68), which, in the case of the interval [–1,1], is 2^{n+1}.

Since by (2.93) the error curve $y = f - p_n(f;\cdot)$ for Chebyshev points (2.95) is essentially *equilibrated* (modulo the variation in the factor $f^{(n+1)}$), and thus free of the violent oscillations we saw for equally spaced points, we would expect more favorable convergence properties for the triangular array (2.64) consisting of Chebyshev nodes. Indeed, one can prove, for example, that

$$p_n(f;\hat{x}_0^{(n)}, \hat{x}_1^{(n)}, \dots, \hat{x}_n^{(n)}; x) \to f(x) \ \text{ as } \ n \to \infty, \tag{2.97}$$

uniformly on [–1,1], provided only that $f \in C^1[-1,1]$. Thus we do not need analyticity of f for (2.97) to hold.

We finally remark – as already suggested by the recurrence relation (2.83) – that Chebyshev polynomials are a special case of orthogonal polynomials. Indeed, the measure in question is precisely (up to an unimportant constant factor) the arc sine measure

$$\mathrm{d}\lambda(x) = \frac{\mathrm{d}x}{\sqrt{1-x^2}} \ \text{ on } \ [-1,1] \tag{2.98}$$

already mentioned in Example 2 of Sect. 2.1.4. This is easily verified from (2.81) and the orthogonality of the cosines (cf. Sect. 2.1.4, (2.33)):

$$\int_{-1}^{1} T_k(x)T_\ell(x)\frac{\mathrm{d}x}{\sqrt{1-x^2}} = \int_0^\pi T_k(\cos\theta)T_\ell(\cos\theta)\mathrm{d}\theta$$

$$= \int_0^\pi \cos k\theta \cos \ell\theta \mathrm{d}\theta = \begin{cases} 0 & \text{if} \quad k \ne \ell, \\ \pi & \text{if} \quad k = \ell = 0, \\ \frac{1}{2}\pi & \text{if} \quad k = \ell > 0. \end{cases} \tag{2.99}$$

The Fourier expansion in Chebyshev polynomials (essentially the Fourier cosine expansion) is therefore given by

$$f(x) = {\sum_{j=1}^{\infty}}' c_j T_j(x) = \frac{1}{2}c_0 + \sum_{j=1}^{\infty} c_j T_j(x), \tag{2.100}$$

where

$$c_j = \frac{2}{\pi}\int_{-1}^{1} f(x)T_j(x)\frac{\mathrm{d}x}{\sqrt{1-x^2}}, \quad j = 0, 1, 2, \ldots. \tag{2.101}$$

Truncating (2.100) with the term of degree n gives a useful polynomial approximation of degree n,

$$\tau_n(x) = {\sum_{j=0}^{n}}' c_j T_j(x),$$

having an error

$$f(x) - \tau_n(x) = \sum_{j=n+1}^{\infty} c_j T_j(x) \approx c_{n+1}T_{n+1}(x). \tag{2.102}$$

The approximation on the far right is better the faster the Fourier coefficients c_j tend to zero. The error (2.102), therefore, essentially oscillates between $+c_{n+1}$ and $-c_{n+1}$ as x varies on the interval [–1,1], and thus is of "uniform" size. This is in stark contrast to Taylor's expansion at $x = 0$, where the nth-degree partial sum has an error proportional to x^{n+1} on [–1,1].

2.2.5 Barycentric Formula

Lagrange's formula (2.52) is attractive more for theoretical purposes than for practical computational work. It can be rewritten, however, in a form that makes it efficient computationally, and that also allows additional interpolation nodes to be added with ease. Having the latter feature in mind, we now assume a sequential set $x_0, x_1, x_2, \ldots$ of interpolation nodes and denote by $p_n(f;\cdot)$ the polynomial of degree $\le n$ interpolating to f at the first $n+1$ of them. We do not assume that the x_i are in any particular order, as long as they are mutually distinct.

We introduce a triangular array of auxiliary quantities defined by

$$\lambda_0^{(0)} = 1, \ \lambda_i^{(n)} = \prod_{\substack{j=0 \\ j\ne i}}^{n} \frac{1}{x_i - x_j}, \quad i = 0, 1, \ldots, n; \quad n = 1, 2, 3, \ldots. \tag{2.103}$$

The elementary Lagrange interpolation polynomials of degree n, (2.49), can then be written in the form

$$\ell_i(x) = \frac{\lambda_i^{(n)}}{x - x_i}\,\omega_n(x), \ i = 0, 1, \ldots, n; \ \ \omega_n(x) = \prod_{j=0}^{n}(x - x_j). \tag{2.104}$$

Dividing Lagrange's formula through by $1 \equiv \sum_{i=0}^{n} \ell_i(x)$, one finds

$$p_n(f;x) = \sum_{i=0}^{n} f_i \ell_i(x) = \frac{\displaystyle\sum_{i=0}^{n} f_i \ell_i(x)}{\displaystyle\sum_{i=0}^{n} \ell_i(x)} = \frac{\displaystyle\sum_{i=0}^{n} f_i \frac{\lambda_i^{(n)}}{x-x_i}\omega_n(x)}{\displaystyle\sum_{i=0}^{n} \frac{\lambda_i^{(n)}}{x-x_i}\omega_n(x)},$$

that is,

$$p_n(f;x) = \frac{\displaystyle\sum_{i=0}^{n} \frac{\lambda_i^{(n)}}{x-x_i} f_i}{\displaystyle\sum_{i=0}^{n} \frac{\lambda_i^{(n)}}{x-x_i}}, \quad x \neq x_i \text{ for } i = 0, 1, \ldots, n. \tag{2.105}$$

This expresses the interpolation polynomial as a weighted average of the function values $f_i = f(x_i)$ and is, therefore, called the *barycentric formula* – a slight misnomer, since the weights are not necessarily all positive. The auxiliary quantities $\lambda_i^{(n)}$ involved in (2.105) are those in the row numbered n of the triangular array (2.103). Once they have been calculated, the evaluation of $p_n(f;x)$ by (2.105), for any fixed x, is straightforward and cheap. Note, however, that when x is sufficiently close to some x_i, the right-hand side of (2.105) should be replaced by f_i.

Comparison with (2.52) shows that

$$\ell_i(x) = \frac{\dfrac{\lambda_i^{(n)}}{x-x_i}}{\displaystyle\sum_{j=0}^{n} \frac{\lambda_j^{(n)}}{x-x_j}}, \quad i = 0, 1, \ldots, n. \tag{2.106}$$

In order to arrive at an efficient algorithm for computing the required quantities $\lambda_i^{(n)}$, we first note that, for $k \geq 1$,

$$\lambda_i^{(k)} = \frac{\lambda_i^{(k-1)}}{x_i - x_k}, \quad i = 0, 1, \ldots, k-1. \tag{2.107}$$

The last quantity $\lambda_k^{(k)}$ missing in (2.107) is best computed directly from the definition (2.103),

$$\lambda_k^{(k)} = \frac{1}{\prod_{j=0}^{k-1}(x_k - x_j)}, \quad k \geq 1.$$

We thus arrive at the following algorithm:

$$
\begin{aligned}
&\lambda_0^{(0)} = 1,\\
&\text{for } k = 1, 2, \ldots, n \ \text{ do}\\
&\left[\begin{array}{l}
\lambda_i^{(k)} = \dfrac{\lambda_i^{(k-1)}}{x_i - x_k}, \quad i = 0, 1, \ldots, k-1,\\[2ex]
\lambda_k^{(k)} = \dfrac{1}{\prod_{j=0}^{k-1}(x_k - x_j)}.
\end{array}\right.
\end{aligned}
\tag{2.108}
$$

This requires $\frac{1}{2}n(n+1)$ subtractions, $\frac{1}{2}(n-1)n$ multiplications, and $\frac{1}{2}n(n+3)$ divisions for computing the $n+1$ quantities $\lambda_0^{(n)}, \lambda_1^{(n)}, \ldots, \lambda_n^{(n)}$ in (2.105). Therefore, (2.106) in combination with (2.108) is more efficient than (2.49), which requires $O(n^3)$ operations to evaluate. It is also quite stable, since only benign arithmetic operations are involved (disregarding the formation of differences such as $x - x_i$, which occur in both formulae).

If we decide to incorporate the next data point (x_{n+1}, f_{n+1}), all we need to do is extend the k-loop in (2.108) through $n+1$, that is, generate the next row of auxiliary quantities $\lambda_0^{(n+1)}, \lambda_1^{(n+1)}, \ldots, \lambda_{n+1}^{(n+1)}$. We are then ready to compute $p_{n+1}(f; x)$ from (2.105) with n replaced by $n+1$.

2.2.6 *Newton's*[11] *Formula*

This is another way of organizing the work in Sect. 2.2.5. Although the computational effort remains essentially the same, it becomes easier to treat "confluent" interpolation points, that is, multiple points in which not only the function values, but also consecutive derivative values, are given (cf. Sect. 2.2.7).

Using the same setup as in Sect. 2.2.5, we denote

$$p_n(x) = p_n(f; x_0, x_1, \ldots, x_n; x), \quad n = 0, 1, 2, \ldots. \tag{2.109}$$

---

[11] Sir Isaac Newton (1643–1727) was an eminent figure of seventeenth century mathematics and physics. Not only did he lay the foundations of modern physics, but he was also one of the coinventors of differential calculus. Another was Leibniz, with whom he became entangled in a bitter and life-long priority dispute. His most influential work was the *Philosophiae Naturalis Principia Mathematica*, often called simply the *Principia*, one of the greatest work on physics and astronomy ever written. Therein one finds not only his ideas on interpolation, but also his suggestion to use the interpolating polynomial for purposes of integration (cf. Chap. 3, Sect. 3.2.2).

We clearly have

$$
\begin{aligned}
p_0(x) &= a_0, \\
p_n(x) &= p_{n-1}(x) + a_n(x - x_0)(x - x_1)\cdots(x - x_{n-1}), \\
n &= 1, 2, 3, \ldots,
\end{aligned} \tag{2.110}
$$

for some constants a_0, a_1, a_2, ... This gives rise to a new form of the interpolation polynomial,

$$
\begin{aligned}
p_n(f; x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) \\
+ \cdots + a_n(x - x_0)(x - x_1)\cdots(x - x_{n-1}),
\end{aligned} \tag{2.111}
$$

which is called *Newton's form*. The constants involved can be determined, in principle, by the interpolation conditions

$$
\begin{aligned}
f_0 &= a_0, \\
f_1 &= a_0 + a_1(x_1 - x_0), \\
f_2 &= a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1),
\end{aligned}
$$

and so on, which represent a triangular, nonsingular (why?) system of linear algebraic equations. This uniquely determines the constants; for example,

$$
\begin{aligned}
a_0 &= f_0, \\
a_1 &= \frac{f_1 - f_0}{x_1 - x_0}, \\
a_2 &= \frac{f_2 - a_0 - a_1(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)},
\end{aligned}
$$

and so on. Evidently, a_n is a linear combination of f_0, $f_1, \ldots, f_n$, with coefficients that depend on x_0, $x_1, \ldots, x_n$. We use the notation

$$
a_n = [x_0, x_1, \ldots, x_n]f, \quad n = 0, 1, 2, \ldots, \tag{2.112}
$$

for this linear combination, and call the right-hand side the nth *divided difference* of f relative to the nodes x_0, $x_1, \ldots, x_n$. Considered as a function of these $n + 1$ variables, the divided difference is a *symmetric function*; that is, permuting the variables in any way does not affect the value of the function. This is a direct consequence of the fact that a_n in (2.111) is the *leading coefficient of* $p_n(f; x)$: the interpolation polynomial $p_n(f; \cdot)$ surely does not depend on the order in which we write down the interpolation conditions.

The name "divided difference" comes from the useful property

$$[x_0, x_1, x_2, \dots, x_k]f = \frac{[x_1, x_2, \dots, x_k]f - [x_0, x_1, \dots, x_{k-1}]f}{x_k - x_0} \tag{2.113}$$

expressing the kth divided difference as a difference of $(k-1)$st divided differences, divided by a difference of the x_i. Since we have symmetry, the order in which the variables are written down is immaterial; what is important is that the two divided differences (of the same order $k-1$) in the numerator have $k-1$ of the x_i in common. The "extra" one in the first term, and the "extra" one in the second, are precisely the x_i that appear in the denominator, in the same order.

To prove (2.113), let

$$r(x) = p_{k-1}(f; x_1, x_2, \dots, x_k; x)$$

and

$$s(x) = p_{k-1}(f; x_0, x_1, \dots, x_{k-1}; x).$$

Then

$$p_k(f; x_0, x_1, \dots, x_k; x) = r(x) + \frac{x - x_k}{x_k - x_0}[r(x) - s(x)]. \tag{2.114}$$

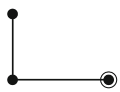
Indeed, the polynomial on the right-hand side has clearly degree $\le k$ and takes on the correct value f_i at x_i, $i = 0, 1, \dots, k$. For example, if $i \ne 0$ and $i \ne k$,

$$r(x_i) + \frac{x_i - x_k}{x_k - x_0}[r(x_i) - s(x_i)] = f_i + \frac{x_i - x_k}{x_k - x_0}[f_i - f_i] = f_i,$$

and similarly for $i = 0$ and for $i = k$. By uniqueness of the interpolation polynomial, this implies (2.114). Now equating the leading coefficients on both sides of (2.114) immediately gives (2.113).

Equation (2.113) can be used to generate the *table of divided differences*:

$$\begin{array}{cc|lll}
x & f & & & \\
\hline
x_0 & f_0 & & & \\
x_1 & f_1 & [x_0, x_1]f & & \\
x_2 & f_2 & [x_1, x_2]f & [x_0, x_1, x_2]f & \\
x_3 & f_3 & [x_2, x_3]f & [x_1, x_2, x_3]f & [x_0, x_1, x_2, x_3]f \\
. & . & \dots & \dots & \dots
\end{array} \tag{2.115}$$



The divided differences are here arranged in such a manner that their computation proceeds according to one single rule: *each entry is the difference of the entry immediately to the left and the one above it, divided by the difference of the x-value* *horizontally to the left and the one opposite the f-value found by going diagonally up*. Each entry, therefore, is calculated from its two neighbors immediately to the left, which is expressed by the computing stencil in (2.115).

The divided differences $a_0, a_1, \dots, a_n$ (cf. (2.112)) that occur in Newton's formula (2.111) are precisely the first $n+1$ *diagonal entries* in the table of divided differences. Their computation requires $n(n+1)$ additions and $\frac{1}{2}n(n+1)$ divisions, essentially the same effort that was required in computing the auxiliary quantities $\lambda_i^{(n)}$ in the barycentric formula (cf. Ex. 61). Adding another data point (x_{n+1}, f_{n+1}) requires the generation of the next line of divided differences. The last entry of this line is a_{n+1}, and we can update $p_n(f;x)$ by adding to it the term $a_{n+1}(x-x_0)(x-x_1)\cdots(x-x_n)$ to get p_{n+1} (cf. (2.110)).

Example.

x	f			
0	3			
1	4	(4–3)/(1–0) = 1		
2	7	(7–4)/(2–1) = 3	(3–1)/(2–0) = 1	
4	19	(19–7)/(4–2) = 6	(6–3)/(4–1) = 1	(1–1)/(4–0) = 0

The cubic interpolation polynomial is

$$\begin{aligned} p_3(f;x) &= 3 + 1\cdot(x-0) + 1\cdot(x-0)(x-1) + 0\cdot(x-0)(x-1)(x-2) \\ &= 3 + x + x(x-1) = 3 + x^2, \end{aligned}$$

which indeed is the function tabulated. Note that the leading coefficient of $p_3(f;\cdot)$ is zero, which is why the last divided difference turned out to be 0.

Newton's formula also yields a new representation for the error term in Lagrange interpolation. Let t temporarily denote an arbitrary "node" not equal to any of the $x_0, x_1, \dots, x_n$. Then we have,

$$\begin{aligned} &p_{n+1}(f;x_0,x_1,\dots,x_n,t;x) \\ &\quad = p_n(f;x) + [x_0,x_1,\dots,x_n,t]f\cdot\prod_{i=0}^{n}(x-x_i). \end{aligned}$$

Now put $x = t$; since the polynomial on the left-hand side interpolates to f at t, we get

$$f(t) = p_n(f;t) + [x_0,x_1,\dots,x_n,t]f\cdot\prod_{i=0}^{n}(t-x_i).$$

Writing again x for t (which was arbitrary, after all), we find

$$f(x) - p_n(f;x) = [x_0,x_1,\dots,x_n,x]f\cdot\prod_{i=0}^{n}(x-x_i). \tag{2.116}$$

This is the new formula for the interpolation error. Note that it involves no derivative of f, only function values. The trouble is, that $f(x)$ is one of them. Indeed, (2.116) is basically a tautology since, when everything is written out explicitly, the formula evaporates to $0 = 0$, which is correct, but not overly exciting.

In spite of this seeming emptiness of (2.116), we can draw from it an interesting and very useful conclusion. (For another application, see Chap. 3, Ex. 2.) Indeed, compare it with the earlier formula (2.60); one obtains

$$[x_0, x_1, \ldots, x_n, x]f = \frac{f^{(n+1)}(\xi(x))}{(n+1)!},$$

where $x_0, x_1, \ldots, x_n, x$ are arbitrary distinct points in $[a, b]$ and $f \in C^{n+1}[a, b]$. Moreover, $\xi(x)$ is strictly between the smallest and largest of these points (cf. the proof of (2.60)). We can now write $x = x_{n+1}$, and then replace $n + 1$ by n to get

$$[x_0, x_1, \ldots, x_n]f = \frac{1}{n!} f^{(n)}(\xi). \tag{2.117}$$

Thus, for any $n + 1$ distinct points in $[a, b]$ and any $f \in C^n[a, b]$, *the divided difference of f of order n is the* nth *scaled derivative of f at some* (unknown) *intermediate point*. If we now let all x_i, $i \geq 1$, tend to x_0, then ξ, being trapped between them, must also tend to x_0, and, since $f^{(n)}$ is continuous at x_0, we obtain

$$[\underbrace{x_0, x_0, \ldots, x_0}_{n+1 \text{ times}}]f = \frac{1}{n!} f^{(n)}(x_0). \tag{2.118}$$

This suggests that *the* nth *divided difference at* n+1 *"confluent"* (i.e., identical) *points be defined to be the* nth *derivative at this point divided by* n!. This allows us, in the next section, to solve the Hermite interpolation problem.

2.2.7 Hermite[12] Interpolation

The general Hermite interpolation problem consists of the following: given $K + 1$ distinct points $x_0, x_1, \ldots, x_K$ in $[a, b]$ and corresponding integers $m_k \geq 1$, and given a function $f \in C^{M-1}[a, b]$, with $M = \max_k m_k$, find a polynomial p of lowest degree such that, for $k = 0, 1, \ldots, K$,

$$p^{(\mu)}(x_k) = f_k^{(\mu)}, \quad \mu = 0, 1, \ldots, m_k - 1, \tag{2.119}$$

where $f_k^{(\mu)} = f^{(\mu)}(x_k)$ is the μth derivative of f at x_k.

[12]Charles Hermite (1822–1901) was a leading French mathematician. An Academician in Paris, known for his extensive work in number theory, algebra, and analysis, he is famous for his proof in 1873 of the transcendental nature of the number e. He was also a mentor of the Dutch mathematician Stieltjes.

The problem can be thought of as a limiting case of Lagrange interpolation if we consider x_k to be a *point of multiplicity* m_k, that is, obtained by a confluence of m_k distinct points into a single point x_k. We can imagine setting up the table of divided differences, and Newton's interpolation formula, just before the confluence takes place, and then simply "go to the limit." To do this in practice requires that each point x_k be entered exactly m_k times in the first column of the table of divided differences. The formula (2.118) then allows us to *initialize* the divided differences for these points. For example, if $m_k = 4$, then

$$\begin{array}{cccccc} x & f & & & \\ \hline \cdot & \cdot & \cdot & \cdot & \cdot \\ x_k & f_k & & & \\ x_k & f_k & f'_k & & \\ x_k & f_k & f'_k & \frac{1}{2}f''_k & \\ x_k & f_k & f'_k & \frac{1}{2}f''_k & \frac{1}{6}f'''_k \\ \cdot & \cdot & \cdot & \cdot & \cdot \end{array} \tag{2.120}$$

Doing this initialization for each k, we are then ready to complete the table of divided differences in the usual way. (There will be no zero divisors; they have been taken care of during the initialization.) We obtain a table with $m_0 + m_1 + \cdots + m_K$ entries in the first column, and hence an interpolation polynomial of degree $\leq n = m_0 + m_1 + \cdots + m_K - 1$, which, as in the Lagrange case, is unique. The $n + 1$ diagonal entries in the table give us the coefficients in Newton's formula, as before, except that in the product terms of the formula, some of the factors are repeated. Also the error term of interpolation remains in force, with the repetition of factors properly accounted for.

We illustrate the procedure with two simple examples.

1. Find $p \in \mathbb{P}_3$ such that

$$p(x_0) = f_0,\ p'(x_0) = f'_0,\ p''(x_0) = f''_0,\ p'''(x_0) = f'''_0.$$

Here $K = 0$, $m_0 = 4$, that is, we have a single quadruple point. The table of divided differences is precisely the one in (2.120) (with $k = 0$); hence Newton's formula becomes

$$p(x) = f_0 + (x - x_0)f'_0 + \frac{1}{2}(x - x_0)^2 f''_0 + \frac{1}{6}(x - x_0)^3 f'''_0,$$

which is nothing but the Taylor polynomial of degree 3. Thus Taylor's polynomial is a special case of a Hermite interpolation polynomial. The error term of interpolation, furthermore, gives us

$$f(x) - p(x) = \frac{1}{24}(x - x_0)^4 f^{(4)}(\xi), \quad \xi \text{ between } x_0 \text{ and } x,$$

which is Lagrange's form of the remainder term in Taylor's formula.

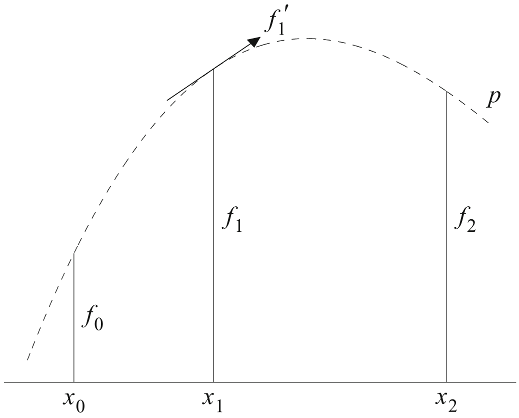


Fig. 2.8 A Hermite interpolation problem

2. Find $p \in \mathbb{P}_3$ such that

$$p(x_0) = f_0, \quad p(x_1) = f_1, \quad p'(x_1) = f_1', \quad p(x_2) = f_2,$$

where $x_0 < x_1 < x_2$ (cf. Fig. 2.8).

The table of divided differences now has the form:

x	f			
x_0	f_0			
x_1	f_1	$[x_0, x_1]f$		
x_1	f_1	f_1'	$[x_0, x_1, x_1]f$	
x_2	f_2	$[x_1, x_2]f$	$[x_1, x_1, x_2]f$	$[x_0, x_1, x_1, x_2]f$.

If we denote the diagonal entries, as before, by a_0, a_1, a_2, a_3, Newton's formula takes the form

$$p(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + a_3(x - x_0)(x - x_1)^2,$$

and the error formula becomes

$$f(x) - p(x) = (x - x_0)(x - x_1)^2(x - x_2)\frac{f^{(4)}(\xi)}{4!}, \quad x_0 < \xi < x_2.$$

For equally spaced points, say, $x_0 = x_1 - h$, $x_2 = x_1 + h$, we have, if $x = x_1 + th$, $-1 \le t \le 1$,

$$|(x - x_0)(x - x_1)^2(x - x_2)| = |(t^2 - 1)t^2 \cdot h^4| \le \tfrac{1}{4}h^4,$$

and so

$$\|f - p\|_\infty \le \frac{1}{4} h^4 \frac{\|f^{(4)}\|_\infty}{24} = \frac{h^4}{96} \|f^{(4)}\|_\infty,$$

with the ∞-norm referring to the interval $[x_0, x_2]$.

2.2.8 Inverse Interpolation

An interesting application of interpolation – and, in particular, of Newton's formula – is to the solution of a nonlinear equation,

$$f(x) = 0. \tag{2.121}$$

Here f is a given (nonlinear) function, and we are interested in a root α of the equation for which we already have two approximations,

$$x_0 \approx \alpha, \quad x_1 \approx \alpha.$$

We assume further that near the root α, the function f is monotone, so that

$$y = f(x) \text{ has an inverse } x = f^{-1}(y).$$

Denote, for short,

$$g(y) = f^{-1}(y).$$

Since $\alpha = g(0)$, our problem is to evaluate $g(0)$. From our two approximations, we can compute $y_0 = f(x_0)$ and $y_1 = f(x_1)$, giving $x_0 = g(y_0)$, $x_1 = g(y_1)$. Hence, we can start a table of divided differences for the inverse function g:

y	g	
y_0	x_0	
y_1	x_1	$[y_0, y_1]g$

Wanting to compute $g(0)$, we can get a first improved approximation by linear interpolation,

$$x_2 = x_0 + (0 - y_0)[y_0, y_1]g = x_0 - y_0[y_0, y_1]g.$$

Now evaluating $y_2 = f(x_2)$, we get $x_2 = g(y_2)$. Hence, the table of divided differences can be updated and becomes

y	g		
y_0	x_0		
y_1	x_1	$[y_0, y_1]g$	
y_2	x_2	$[y_1, y_2]g$	$[y_0, y_1, y_2]g$

This allows us to use quadratic interpolation to get, again with Newton's formula,

$$x_3 = x_2 + (0 - y_0)(0 - y_1)[y_0, y_1, y_2]g = x_2 + y_0 y_1 [y_0, y_1, y_2]g$$

and then

$$y_3 = f(x_3), \quad \text{and} \quad x_3 = g(y_3).$$

Since y_0, y_1 are small, the product $y_0 y_1$ is even smaller, making the correction term added to the linear interpolant x_2 quite small. If necessary, we can continue updating the difference table,

y	g			
y_0	x_0			
y_1	x_1	$[y_0, y_1]g$		
y_2	x_2	$[y_1, y_2]g$	$[y_0, y_1, y_2]g$	
y_3	x_3	$[y_2, y_3]g$	$[y_1, y_2, y_3]g$	$[y_0, y_1, y_2, y_3]g$

and computing

$$x_4 = x_3 - y_0 y_1 y_2 [y_0, y_1, y_2, y_3]g, \quad y_4 = f(x_4), \quad x_4 = g(y_4),$$

giving us another data point to generate the next row of divided differences, and so on. In general, the process will converge rapidly: $x_k \to \alpha$ as $k \to \infty$. The precise analysis of convergence, however, is not simple because of the complicated structure of the successive derivatives of the inverse function $g = f^{-1}$.

2.3 Approximation and Interpolation by Spline Functions

Our concern in Sect. 2.1.1 was with approximation of functions by a single polynomial over a finite interval $[a,b]$. When more accuracy was wanted, we simply increased the degree of the polynomial, and under suitable assumptions the approximation indeed can be made as accurate as one wishes by choosing the degree of the approximating polynomial sufficiently large.

However, there are other ways to control accuracy. One is to impose a subdivision Δ upon the interval $[a,b]$,

$$\Delta: \quad a = x_1 < x_2 < x_3 < \cdots < x_{n-1} < x_n = b, \tag{2.122}$$

and use *low-degree* polynomials on each subinterval $[x_i, x_{i+1}]$ $(i = 1, 2, \ldots, n-1)$ to approximate the given function. The rationale behind this is the recognition that on a sufficiently small interval, functions can be approximated arbitrarily well by polynomials of low degree, even degree 1, or zero, for that matter. Thus, measuring the "fineness" of the subdivision Δ by

$$|\Delta| = \max_{1 \le i \le n-1} \Delta x_i, \quad \Delta x_i = x_{i+1} - x_i, \tag{2.123}$$

we try to control (increase) the accuracy by varying (decreasing) $|\Delta|$, keeping the degrees of the polynomial pieces uniformly low.

To discuss these approximation processes, we make use of the class of functions (cf. Example 2 at the beginning of Chap. 2)

$$\mathbb{S}_m^k(\Delta) = \left\{ s : s \in C^k[a,b], \ s\big|_{[x_i, x_{i+1}]} \in \mathbb{P}_m, \ i = 1, 2, \ldots, n-1 \right\}, \tag{2.124}$$

where $m \ge 0, k \ge 0$ are given nonnegative integers. We refer to $\mathbb{S}_m^k(\Delta)$ as the *spline functions of degree m and smoothness class k* relative to the subdivision Δ. (If the subdivision is understood from the context, we omit Δ in the notation on the left-hand side of (2.124).) The point in the continuity assumption of (2.124), of course, is that the kth derivative of s is to be continuous *everywhere* on $[a,b]$, in particular, also at the subdivision points x_i $(i = 2, \ldots, n-1)$ of Δ. One extreme case is $k = m$, in which case $s \in \mathbb{S}_m^m$ necessarily consists of just one single polynomial of degree m on the whole interval $[a,b]$; that is, $\mathbb{S}_m^m = \mathbb{P}_m$ (see Ex. 68). Since we want to get away from $\mathbb{P}_m$, we assume $k < m$. The other extreme is the case where no continuity at all (at the subdivision points x_i) is required; we then put $k = -1$. Thus $\mathbb{S}_m^{-1}(\Delta)$ is the class of piecewise polynomials of degree $\le m$, where the polynomial pieces can be completely disjoint (see Fig. 2.9).

We begin with the simplest case – piecewise linear approximation – that is, the case $m = 1$ (hence $k = 0$).

2.3.1 Interpolation by Piecewise Linear Functions

The problem here is to find an $s \in \mathbb{S}_1^0(\Delta)$ such that, for a given function f defined on $[a,b]$, we have

$$s(x_i) = f_i \quad \text{where} \quad f_i = f(x_i), \ \ i = 1, 2, \ldots, n. \tag{2.125}$$

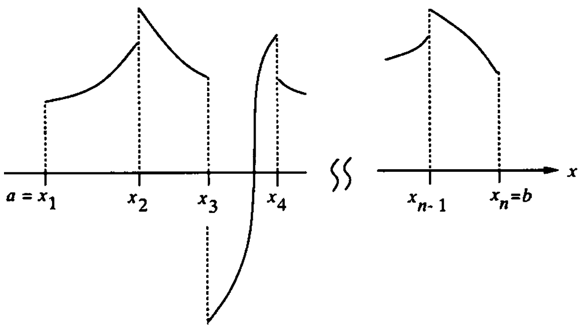


Fig. 2.9 A function $s \in \mathbb{S}_m^{-1}$

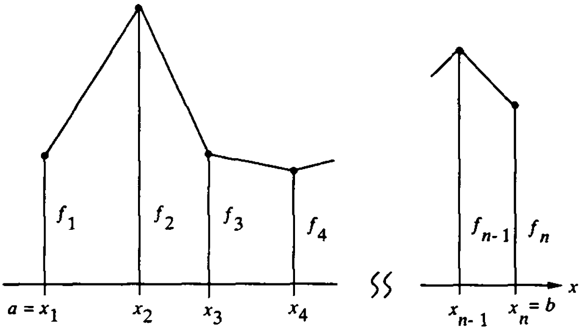


Fig. 2.10 Piecewise linear interpolation.

We conveniently let the interpolation nodes coincide with the points x_i of the subdivision Δ in (2.122). This simplifies matters, but is not necessary (cf. Ex. 75). The solution then indeed is trivial; see Fig. 2.10. If we denote the (obviously unique) interpolant by $s(\cdot) = s_1(f;\cdot)$, then the formula of linear interpolation gives

$$s_1(f;x) = f_i + (x - x_i)[x_i, x_{i+1}]f \quad \text{for } x_i \le x \le x_{i+1}, \quad i = 1, 2, \ldots, n-1. \tag{2.126}$$

A bit more interesting is the analysis of the error. This, too, however, is quite straightforward, once we note that $s_1(f;\cdot)$ on $[x_i, x_{i+1}]$ is simply the linear interpolant to f. Thus, from the theory of (linear) interpolation,

$$f(x) - s_1(f;x) = (x - x_i)(x - x_{i+1})[x_i, x_{i+1}, x]f \quad \text{for } x \in [x_i, x_{i+1}];$$

hence, if $f \in C^2[a,b]$,

$$|f(x) - s_1(f;x)| \le \frac{(\Delta x_i)^2}{8} \max_{[x_i, x_{i+1}]} |f''|, \quad x \in [x_i, x_{i+1}].$$

It then follows immediately that

$$\| f(\cdot) - s_1(f;\cdot)\|_\infty \le \tfrac{1}{8}\,|\Delta|^2 \| f''\|_\infty, \tag{2.127}$$

where the maximum norms are those on $[a,b]$; that is, $\|g\|_\infty = \max_{[a,b]} |g|$. This shows that the error indeed can be made arbitrarily small, uniformly on $[a,b]$, by taking $|\Delta|$ sufficiently small. Making $|\Delta|$ smaller, of course, increases the number of polynomial pieces, and with it, the volume of data.

It is easy to show (see Ex. 80(b)) that

$$\mathrm{dist}_\infty(f, \mathbb{S}_1^0) \le \| f(\cdot) - s_1(f;\cdot)\|_\infty \le 2\,\mathrm{dist}_\infty(f, \mathbb{S}_1^0), \tag{2.128}$$

where, for any set of functions $\mathbb{S}$,

$$\mathrm{dist}_\infty(f, \mathbb{S}) := \inf_{s\in\mathbb{S}} \| f - s\|_\infty.$$

In other words, the piecewise linear interpolant $s_1(f;\cdot)$ is a nearly optimal approximation, its error differing from the error of the best approximant to f from $\mathbb{S}_1^0$ by at most a factor of 2.

2.3.2 A Basis for $\mathbb{S}_1^0(\Delta)$

What is the dimension of the space $\mathbb{S}_1^0(\Delta)$? In other words, how many degrees of freedom do we have? If, for the moment, we ignore the continuity requirement (i.e., if we look at $\mathbb{S}_1^{-1}(\Delta)$), then each linear piece has two degrees of freedom, and there are $n-1$ pieces; so $\dim \mathbb{S}_1^{-1}(\Delta) = 2n-2$. Each continuity requirement imposes one equation, and hence reduces the degree of freedom by 1. Since continuity must be enforced only at the interior subdivision points x_i, $i = 2, \dots, n-1$, we find that $\dim \mathbb{S}_1^0(\Delta) = 2n-2-(n-2) = n$. So we expect that a basis of $\mathbb{S}_1^0(\Delta)$ must consist of exactly n basis functions.

We now define n such functions. For notational convenience, we let $x_0 = x_1$ and $x_{n+1} = x_n$; then, for $i = 1, 2, \dots, n$, we define

$$B_i(x) = \begin{cases} \dfrac{x - x_{i-1}}{x_i - x_{i-1}} & \text{if } x_{i-1} \le x \le x_i, \\[2ex] \dfrac{x_{i+1} - x}{x_{i+1} - x_i} & \text{if } x_i \le x \le x_{i+1}, \\[2ex] 0 & \text{otherwise.} \end{cases} \tag{2.129}$$

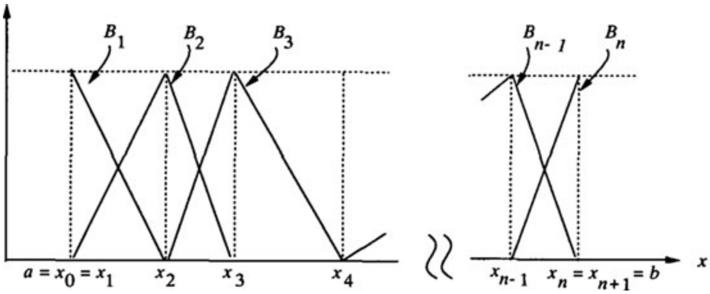


Fig. 2.11 The functions B_i

Note that the first equation, when $i = 1$, and the second, when $i = n$, are to be ignored, since x in both cases is restricted to a single point and the ratio in question has the meaningless form 0/0. (It is the other ratio that provides the necessary information in these cases.) The functions B_i may be referred to as "hat functions" (Chinese hats), but note that the first and last hat is cut in half. The functions B_i are depicted in Fig. 2.11. We expect these functions to form a basis of $\mathbb{S}_1^0(\Delta)$. To prove this, we must show:

(a) the functions $\{B_i\}_{i=1}^n$ are linearly independent and
(b) they span the space $\mathbb{S}_1^0(\Delta)$.

Both these properties follow from the basic fact that

$$B_i(x_j) = \delta_{ij} = \begin{cases} 1 & \text{if} \quad i = j, \\ 0 & \text{if} \quad i \neq j, \end{cases} \tag{2.130}$$

which one easily reads from Fig. 2.11. To show (a), assume there is a linear combination of the B_i that vanishes identically on $[a,b]$,

$$s(x) = \sum_{i=1}^{n} c_i B_i(x), \quad s(x) \equiv 0 \text{ on } [a, b]. \tag{2.131}$$

Putting $x = x_j$ in (2.131) and using (2.130) then gives $c_j = 0$. Since this holds for each $j = 1, 2, \ldots, n$, we see that only the trivial linear combination (with all $c_i = 0$) can vanish identically. To prove (b), let $s \in \mathbb{S}_1^0(\Delta)$ be given arbitrarily. We must show that s can be represented as a linear combination of the B_i. We claim that, indeed,

$$s(x) = \sum_{i=1}^{n} s(x_i) B_i(x). \tag{2.132}$$

This is so, because the function on the right-hand side has the same values as s at each x_j, and therefore, being in $\mathbb{S}_1^0(\Delta)$, must coincide with s.

Equation (2.132), which holds for every $s \in \mathbb{S}_1^0(\Delta)$, may be thought of as the analogue of the Lagrange interpolation formula for polynomials. The role of the elementary Lagrange polynomials ℓ_i is now played by the B_i.

2.3.3 Least Squares Approximation

As an application of the basis $\{B_i\}$, we consider the problem of least squares approximation on $[a,b]$ by functions in $\mathbb{S}_1^0(\Delta)$. The discrete L_2 approximation problem with data given at the points x_i $(i = 1, 2, \ldots, n)$, of course, has the trivial solution $s_1(f;\cdot)$, which drives the error to zero at each data point. We therefore consider only the continuous problem: given $f \in C[a,b]$, find $\hat{s}_1(f;\cdot) \in \mathbb{S}_1^0(\Delta)$ such that

$$\int_a^b [f(x) - \hat{s}_1(f;x)]^2 \mathrm{d}x \le \int_a^b [f(x) - s(x)]^2 \mathrm{d}x \quad \text{for all} \quad s \in \mathbb{S}_1^0(\Delta). \tag{2.133}$$

Writing

$$\hat{s}_1(f;x) = \sum_{i=1}^n \hat{c}_i B_i(x), \tag{2.134}$$

we know from the general theory of Sect. 2.1 that the coefficients $\hat{c}_i$ must satisfy the normal equations

$$\sum_{j=1}^n \left[\int_a^b B_i(x) B_j(x) \mathrm{d}x \right] \hat{c}_j = \int_a^b B_i(x) f(x) \mathrm{d}x, \quad i = 1, 2, \ldots, n. \tag{2.135}$$

Now the fact that B_i is nonzero only on (x_{i-1}, x_{i+1}) implies that $\int_a^b B_i(x) \cdot B_j(x)\mathrm{d}x = 0$ if $|i - j| > 1$; that is, the system (2.135) is tridiagonal. An easy computation (cf. Ex. 77) indeed yields

$$\frac{1}{6}\Delta x_{i-1} \cdot \hat{c}_{i-1} + \frac{1}{3}(\Delta x_{i-1} + \Delta x_i)\hat{c}_i + \frac{1}{6}\Delta x_i \cdot \hat{c}_{i+1} = b_i,\ i = 1, 2, \ldots, n, \tag{2.136}$$

where $b_i = \int_a^b B_i(x) f(x)\mathrm{d}x = \int_{x_{i-1}}^{x_{i+1}} B_i(x) f(x)\mathrm{d}x$. Note, by our convention, that $\Delta x_0 = 0$ and $\Delta x_n = 0$, so that (2.136) is in fact a tridiagonal system for the unknowns $\hat{c}_1, \hat{c}_2, \ldots, \hat{c}_n$. Its matrix is given by

$$\begin{bmatrix} \frac{1}{3}\Delta x_1 & \frac{1}{6}\Delta x_1 & & & 0 \\ \frac{1}{6}\Delta x_1 & \frac{1}{3}(\Delta x_1 + \Delta x_2) & \frac{1}{6}\Delta x_2 & & \\ & \frac{1}{6}\Delta x_2 & & \ddots & \\ & & \ddots & \ddots & \frac{1}{6}\Delta x_{n-1} \\ 0 & & & \frac{1}{6}\Delta x_{n-1} & \frac{1}{3}\Delta x_{n-1} \end{bmatrix}.$$

As it must be, by the general theory of Sect. 2.1, the matrix is symmetric and positive definite, but it is also diagonally dominant, each diagonal element exceeding by a factor of 2 the sum of the (positive) off-diagonal elements in the same row. The system (2.136) can therefore be solved easily, rapidly, and accurately by the Gauss elimination procedure, and there is no need for pivoting.

Like the interpolant $s_1(f;\cdot)$, the least squares approximant $\hat{s}_1(f;\cdot)$, too, can be shown to be nearly optimal, in that

$$\mathrm{dist}_\infty(f, \mathbb{S}_1^0) \le \|f(\cdot) - \hat{s}_1(f;\cdot)\|_\infty \le 4\,\mathrm{dist}_\infty(f, \mathbb{S}_1^0). \tag{2.137}$$

The spread is now by a factor of 4, rather than 2, as in (2.128).

2.3.4 Interpolation by Cubic Splines

The most widely used splines are cubic splines, in particular, cubic spline interpolants. We first discuss the interpolation problem for splines $s \in \mathbb{S}_3^1(\Delta)$. Continuity of the first derivative of any cubic spline interpolant $s_3(f;\cdot)$ can be enforced by prescribing the values of the first derivative at each point x_i, $i = 1, 2, \ldots, n$. Thus let $m_1, m_2, \ldots, m_n$ be arbitrary given numbers, and denote

$$s_3(f;\cdot)|_{[x_i, x_{i+1}]} = p_i(x), \quad i = 1, 2, \ldots, n-1. \tag{2.138}$$

Then, we enforce $s_3'(f; x_i) = m_i$, $i = 1, 2, \ldots, n$, by selecting each piece p_i of $s_3(f;\cdot)$ to be the (unique) solution of a Hermite interpolation problem, namely,

$$\begin{aligned} p_i(x_i) &= f_i, & p_i(x_{i+1}) &= f_{i+1}, \\ p_i'(x_i) &= m_i, & p_i'(x_{i+1}) &= m_{i+1}, \end{aligned} \quad i = 1, 2, \ldots, n-1. \tag{2.139}$$

We solve (2.139) by Newton's interpolation formula. The required divided differences are:

$$\begin{array}{lllll}
x_i & f_i & & & \\
x_i & f_i & m_i & & \\
x_{i+1} & f_{i+1} & [x_i, x_{i+1}]f & \dfrac{[x_i, x_{i+1}]f - m_i}{\Delta x_i} & \\
x_{i+1} & f_{i+1} & m_{i+1} & \dfrac{m_{i+1} - [x_i, x_{i+1}]f}{\Delta x_i} & \dfrac{m_{i+1} + m_i - 2[x_i, x_{i+1}]f}{(\Delta x_i)^2}
\end{array}$$

and the interpolation polynomial (in Newton's form) is

$$\begin{aligned}
p_i(x) = f_i + (x - x_i)m_i + (x - x_i)^2 \, \frac{[x_i, x_{i+1}]f - m_i}{\Delta x_i} \\
+ (x - x_i)^2 (x - x_{i+1}) \, \frac{m_{i+1} + m_i - 2[x_i, x_{i+1}]f}{(\Delta x_i)^2}.
\end{aligned}$$

Alternatively, in Taylor's form, we can write

$$\begin{aligned}
&p_i(x) = c_{i,0} + c_{i,1}(x - x_i) + c_{i,2}(x - x_i)^2 + c_{i,3}(x - x_i)^3, \\
&\quad x_i \le x \le x_{i+1},
\end{aligned} \tag{2.140}$$

where, by noting that $x - x_{i+1} = x - x_i - \Delta x_i$,

$$\begin{aligned}
c_{i,0} &= f_i, \\
c_{i,1} &= m_i, \\
c_{i,2} &= \frac{[x_i, x_{i+1}]f - m_i}{\Delta x_i} - c_{i,3} \cdot \Delta x_i, \\
c_{i,3} &= \frac{m_{i+1} + m_i - 2[x_i, x_{i+1}]f}{(\Delta x_i)^2}.
\end{aligned} \tag{2.141}$$

Thus to compute $s_3(f; x)$ for any given $x \in [a, b]$ that is not an interpolation node, one first locates the interval $[x_i, x_{i+1}]$ containing x and then computes the corresponding piece (2.138) by (2.140) and (2.141).

We now discuss some possible choices of the parameters $m_1, m_2, \ldots, m_n$.

(a) *Piecewise cubic Hermite interpolation*. Here one selects $m_i = f'(x_i)$, assuming that these derivative values are known. This gives rise to a strictly *local* scheme, in that each piece p_i can be determined independently from the others. Furthermore, the error of interpolation is easily estimated, since from the theory of interpolation,

$$f(x) - p_i(x) = (x - x_i)^2 (x - x_{i+1})^2 [x_i, x_i, x_{i+1}, x_{i+1}, x]f, \; x_i \le x \le x_{i+1};$$

hence, if $f \in C^4[a,b]$,

$$|f(x) - p_i(x)| \le \left(\frac{1}{2}\Delta x_i\right)^4 \max_{[x_i, x_{i+1}]} \frac{|f^{(4)}|}{4!}, \quad x_i \le x \le x_{i+1}.$$

There follows:

$$\|f(\cdot) - s_3(f;\cdot)\|_\infty \le \tfrac{1}{384} |\Delta|^4 \|f^{(4)}\|_\infty. \tag{2.142}$$

In the case of equally spaced points x_i, one has $|\Delta| = (b-a)/(n-1)$ and, therefore,

$$\|f(\cdot) - s_3(f;\cdot)\|_\infty = O(n^{-4}) \text{ as } n \to \infty. \tag{2.143}$$

This is quite satisfactory, but note that the derivative of f must be known at each point x_i, and the interpolant is only in $C^1[a,b]$.

As to the derivative values, one could approximate them by the derivatives of $p_2(f; x_{i-1}, x_i, x_{i+1}; x)$ at $x = x_i$, which requires only function values of f, except at the endpoints, where again the derivatives of f are involved, the points $a = x_0 = x_1$ and $b = x_n = x_{n+1}$ being double points (cf. Ex. 78). It can be shown that this degrades the accuracy to $O(|\Delta|^3)$.

(b) *Cubic spline interpolation*. Here we require $s_3(f;\cdot) \in \mathbb{S}_3^2(\Delta)$, that is, continuity of the second derivative. In terms of the pieces (2.138) of $s_3(f;\cdot)$, this means that

$$p''_{i-1}(x_i) = p''_i(x_i), \quad i = 2, 3, \ldots, n-1, \tag{2.144}$$

and translates into a condition for the Taylor coefficients in (2.140), namely,

$$2\, c_{i-1,2} + 6\, c_{i-1,3} \cdot \Delta x_{i-1} = 2\, c_{i,2}, \quad i = 2, 3, \ldots, n-1.$$

Plugging in the explicit values (2.141) for these coefficients, we arrive at the linear system

$$(\Delta x_i) m_{i-1} + 2(\Delta x_{i-1} + \Delta x_i) m_i + (\Delta x_{i-1}) m_{i+1} = b_i, \quad i = 2, 3, \ldots, n-1, \tag{2.145}$$

where

$$b_i = 3\{(\Delta x_i)[x_{i-1}, x_i]f + (\Delta x_{i-1})[x_i, x_{i+1}]f\}. \tag{2.146}$$

These are $n-2$ linear equations in the n unknowns $m_1, m_2, \ldots, m_n$. Once m_1 and m_n have been chosen in some way, the system again becomes tridiagonal in the remaining unknowns and hence is readily solved by Gauss elimination. Here are some possible choices of m_1 and m_n.

(b.1) *Complete splines*: $m_1 = f'(a)$, $m_n = f'(b)$. It is known that for this spline

$$\|f^{(r)}(\cdot) - s^{(r)}(f;\cdot)\|_\infty \le c_r|\Delta|^{4-r}\|f^{(4)}\|_\infty,\ r = 0,1,2,3,\ \text{if } f \in C^4[a,b], \tag{2.147}$$

where $c_0 = \frac{5}{384}$, $c_1 = \frac{1}{24}$, $c_2 = \frac{3}{8}$, and c_3 is a constant depending on the mesh ratio $\frac{|\Delta|}{\min_i \Delta x_i}$. Rather remarkably, the bound for $r = 0$ is only five times larger than the bound (2.142) for the piecewise cubic Hermite interpolant, which requires derivative values of f at all interpolation nodes x_i, not just at the endpoints a and b.

(b.2) *Matching of the second derivatives at the endpoints*: $s_3''(f;a) = f''(a)$, $s_3''(f;b) = f''(b)$. Each of these conditions gives rise to an additional equation, namely,

$$\begin{aligned} 2m_1 + m_2 &= 3[x_1,x_2]f - \frac{1}{2}f''(a)\Delta x_1, \\ m_{n-1} + 2m_n &= 3[x_{n-1},x_n]f + \frac{1}{2}f''(b)\Delta x_{n-1}. \end{aligned} \tag{2.148}$$

One conveniently adjoins the first equation to the top of the system (2.145), and the second to the bottom, thereby preserving the tridiagonal structure of the system.

(b.3) *Natural cubic spline*: $s''(f;a) = s''(f;b) = 0$. This again produces two additional equations, which can be obtained from (2.148) by putting there $f''(a) = f''(b) = 0$. They are adjoined to the system (2.145) as described in (b.2). The nice thing about this spline is that it requires only function values of f – no derivatives! – but the price one pays is a degradation of the accuracy to $O(|\Delta|^2)$ near the endpoints (unless indeed $f''(a) = f''(b) = 0$).

(b.4) *"Not-a-knot spline"* (C. de Boor): here we require $p_1(x) \equiv p_2(x)$ and $p_{n-2}(x) \equiv p_{n-1}(x)$; that is, the first two pieces of the spline should be the same polynomial, and similarly for the last two pieces. In effect, this means that the first interior knot x_2, and the last one x_{n-1}, both are inactive (hence the name). This again gives rise to two supplementary equations expressing continuity of $s_3'''(f;x)$ at $x = x_2$ and $x = x_{n-1}$ (cf. Ex. 79).

2.3.5 Minimality Properties of Cubic Spline Interpolants

The complete and natural splines defined in (b.1) and (b.3) of the preceding section have interesting optimality properties. To formulate them, it is convenient to consider not only the subdivision Δ in (2.122), but also the subdivision

$$\Delta' : \quad a = x_0 = x_1 < x_2 < x_3 < \cdots < x_{n-1} < x_n = x_{n+1} = b, \tag{2.149}$$

in which the endpoints are double knots. This means that whenever we interpolate on Δ', we interpolate not only to function values at all interior points but also to the function as well as first derivative values at the endpoints.

The first of the two theorems relates to the complete cubic spline interpolant, $s_{\text{compl}}(f;\cdot)$.

Theorem 2.3.1. *For any function $g \in C^2[a,b]$ that interpolates f on Δ', there holds*

$$\int_a^b [g''(x)]^2 \mathrm{d}x \geq \int_a^b [s''_{\text{compl}}(f;x)]^2 \mathrm{d}x, \tag{2.150}$$

with equality if and only if $g(\cdot) = s_{\text{compl}}(f;\cdot)$.

Note that $s_{\text{compl}}(f;\cdot)$ in Theorem 2.3.1 also interpolates f on Δ', and among all such interpolants its second derivative has the smallest L_2 norm.

Proof of Theorem 2.3.1. We write (for short) $s_{\text{compl}} = s$. The theorem follows, once we have shown that

$$\int_a^b [g''(x)]^2 \mathrm{d}x = \int_a^b [g''(x) - s''(x)]^2 \mathrm{d}x + \int_a^b [s''(x)]^2 \mathrm{d}x. \tag{2.151}$$

Indeed, this immediately implies (2.150), and equality in (2.150) holds if and only if $g''(x) - s''(x) \equiv 0$, which, integrating twice from a to x and using the interpolation properties of s and g at $x = a$ gives $g(x) \equiv s(x)$.

To complete the proof, note that (2.151) is equivalent to

$$\int_a^b s''(x)[g''(x) - s''(x)]\mathrm{d}x = 0. \tag{2.152}$$

Integrating by parts, we get

$$\begin{aligned} &\int_a^b s''(x)[g''(x) - s''(x)]\mathrm{d}x \\ &\quad = s''(x)[g'(x) - s'(x)]\Big|_a^b - \int_a^b s'''(x)[g'(x) - s'(x)]\mathrm{d}x \\ &\quad = -\int_a^b s'''(x)[g'(x) - s'(x)]\mathrm{d}x, \end{aligned} \tag{2.153}$$

since $s'(b) = g'(b) = f'(b)$, and similarly at $x = a$. But s''' is piecewise constant, so

$$\begin{aligned} &\int_a^b s'''(x)[g'(x) - s'(x)]\mathrm{d}x \\ &\quad = \sum_{\nu=1}^{n-1} s'''(x_\nu + 0) \int_{x_\nu}^{x_{\nu+1}} [g'(x) - s'(x)]\mathrm{d}x \\ &\quad = \sum_{\nu=1}^{n-1} s'''(x_\nu + 0)[g(x_{\nu+1}) - s(x_{\nu+1}) - (g(x_\nu) - s(x_\nu))] = 0, \end{aligned}$$

since both s and g interpolate to f on Δ. This proves (2.152) and hence the theorem. $\square$

For interpolation on Δ, the distinction of being optimal goes to the natural cubic spline interpolant $s_{\text{nat}}(f;\cdot)$. This is the content of the second theorem.

Theorem 2.3.2. *For any function $g \in C^2[a,b]$ that interpolates f on Δ (not Δ'), there holds*

$$\int_a^b [g''(x)]^2 \mathrm{d}x \geq \int_a^b [s''_{nat}(f;x)]^2 \mathrm{d}x, \tag{2.154}$$

with equality if and only if $g(\cdot) = s_{nat}(f;\cdot)$.

The proof of Theorem 2.3.2 is virtually the same as that of Theorem 2.3.1, since (2.153) holds again, this time because $s''(b) = s''(a) = 0$. $\square$

Putting $g(\cdot) = s_{\text{compl}}(f;\cdot)$ in Theorem 2.3.2 immediately gives

$$\int_a^b [s''_{\text{compl}}(f;x)]^2 \mathrm{d}x \geq \int_a^b [s''_{\text{nat}}(f;x)]^2 \mathrm{d}x. \tag{2.155}$$

Therefore, in a sense, the natural cubic spline is the "smoothest" interpolant.

The property expressed in Theorem 2.3.2 is the origin of the name "spline." A spline is a flexible strip of wood used in drawing curves. If its shape is given by the equation $y = g(x)$, $a \leq x \leq b$, and if the spline is constrained to pass through the points (x_i, g_i), then it assumes a form that minimizes the bending energy

$$\int_a^b \frac{[g''(x)]^2 \mathrm{d}x}{(1 + [g'(x)]^2)^3}$$

over all functions g similarly constrained. For slowly varying g ($\|g'\|_\infty \ll 1$), this is nearly the same as the minimum property of Theorem 2.3.2.

2.4 Notes to Chapter 2

There are many excellent texts on the general problem of best approximation as exemplified by (2.1). One that emphasizes uniform approximation by polynomials is Feinerman and Newman [1974]; apart from the basic theory of best polynomial approximation, it also contains no fewer than four proofs of the fundamental theorem of Weierstrass. For approximation in the L_∞ and L_1 norm, which is related to linear programming, a number of constructive methods, notably the Remez algorithms and exchange algorithms, are known, both for polynomial and rational approximation. Early, but still very readable, expositions are given in Cheney [1998] and Rivlin [1981], and more recent accounts in Watson [1980] and Powell [1981]. Nearly-best polynomial and rational approximations are widely used in computer routines for special functions; for a survey of work in this area, up to about 1975, see Gautschi [1975a], and for subsequent work, van der Laan and Temme [1984] and Németh [1992]. Much relevant material is also contained in the books by Luke [1975] and [1977]. The numerical approximation and software for special functions is the subject of Gil et al. [2007]; exhaustive documentation can also be found in Lozier and Olver [1994]. A package for some of the more esoteric functions is described in MacLeod [1996]. For an extensive (and mathematically demanding) treatment of rational approximation, the reader is referred to Petrushev and Popov [1987], and for L_1 approximation, to Pinkus [1989]. Methods of nonlinear approximation, including approximation by exponential sums, are studied in Braess [1986]. Other basic texts on approximation and interpolation are Natanson [1964, 1965, 1965] and Davis [1975] from the 1960s, and the more recent books by DeVore and Lorentz [1993] and its sequel, Lorentz et al. [1996]. A large variety of problems of interpolation and approximation by rational functions (including polynomials) in the complex plane is studied in Walsh [1969]. An example of a linear space Φ containing a denumerable set of nonrational basis functions are the sinc functions – scaled translates of $\frac{\sin \pi t}{\pi t}$. They are of importance in the Shannon sampling and interpolation theory (see, e.g., Zayed [1993]) and are also useful for approximation on infinite or semi-infinite domains in the complex plane; see Stenger [1993], [2000] and Kowalski et al. [1995] for an extensive discussion of this. A reader interested in issues of current interest related to multivariate approximation can get a good start by consulting Cheney [1986].

Rich and valuable sources on polynomials and their numerous properties of interest in applied analysis are Milovanović et al. [1994] and Borwein and Erdélyi [1995]. Spline functions – in name and as a basic tool of approximation – were introduced in 1946 by Schoenberg [1946]; also see Schoenberg [1973]. They have generated enormous interest, owing both to their interesting mathematical theory and practical usefulness. There are now many texts available, treating splines from various points of view. A selected list is Ahlberg et al. [1967], Nürnberger [1989], and Schumaker [2007] for the basic theory, de Boor [2001] and Späth [1995] for more practical aspects including algorithms, Atteia [1992] for an abstract treatment based on Hilbert kernels, Bartels et al. [1987] and Dierckx [1993] for applications to computer graphics and geometric modeling, and Chui [1988], de Boor et al. [1993], and Bojanov et al. [1993] for multivariate splines. The standard text on trigonometric series still is Zygmund [2002] .

Section 2.1. Historically, the least squares principle evolved in the context of discrete linear approximation. The principle was first enunciated by Legendre in 1805 in a treatise on celestial mechanics (Legendre [1805]), although Gauss used it earlier in 1794, but published the method only in 1809 (in a paper also on celestial mechanics). For Gauss's subsequent treatises, published in 1821–1826, see the English translation in Gauss [1995]. The statistical justification of least squares as a minimum variance (unbiased) estimator is due to Gauss. If one were to disregard probabilistic arguments, then, as Gauss already remarked (Goldstine [1977, p. 212]), one could try to minimize the sum of any even (positive) power of the errors, and even let this power go to infinity, in which case one would minimize the maximum error. But by these principles " ... we should be led into the most complicated calculations." Interestingly, Laplace at about the same time also proposed discrete L_1 approximation (under the side condition that all errors add up to zero). A reader interested in the history of least squares may wish to consult the article by Sheynin [1993].

The choice of weights w_i in the discrete L_2 norm $\|\cdot\|_{2,w}$ can be motivated on statistical grounds if one assumes that the errors in the data $f(x_i)$ are uncorrelated and have zero mean and variances σ_i^2; an appropriate choice then is $w_i = \sigma_i^{-2}$.

The discrete problem of minimizing $\|f - \varphi\|_{2,w}$ over functions φ in Φ as given by (2.2) can be rephrased in terms of an overdetermined system of linear equations, $\boldsymbol{P}\boldsymbol{c} = \boldsymbol{f}$, where $\boldsymbol{P} = [\pi_j(x_i)]$ is a rectangular matrix of size $N \times n$, and $\boldsymbol{f} = [f(x_i)]$ the data vector of dimension N. If $\boldsymbol{r} = \boldsymbol{f} - \boldsymbol{P}\boldsymbol{c}$, $\boldsymbol{r} = [r_i]$ denotes the residual vector, one tries to find the coefficient vector $\boldsymbol{c} \in \mathbb{R}^n$ such that $\sum_i w_i r_i^2$ is as small as possible. There is a vast literature dealing with overdetermined systems involving more general (full or sparse) matrices and their solution by the method of least squares. A large arsenal of modern techniques of matrix computation can be brought to bear on this problem; see, for example, Björck [1996] for an extensive discussion. In the special case considered here, the method of (discrete) orthogonal polynomials, however, is more efficient. It has its origin in the work of Chebyshev [1859]; a more contemporary exposition, including computational and statistical issues, is given in Forsythe [1957].

There are interesting variations on the theme of polynomial least squares approximation. One is to minimize $\|f - p\|_{2,\mathrm{d}\lambda}$ among all polynomials in $\mathbb{P}_n$ subject to interpolatory constraints at $m+1$ given points, where $m < n$. It turns out that this can be reduced to an unconstrained least squares problem, but for a different measure $\mathrm{d}\lambda$ and a different function f; cf. Gautschi [1996, Sect. 2.1]. Something similar is true for approximation by rational functions with a prescribed denominator polynomial. A more substantial variation consists in wanting to approximate simultaneously a function f and its first s derivatives. In the most general setting, this would require the minimization of $\int_{\mathbb{R}} \sum_{\sigma=0}^{s} [f^{(\sigma)}(t) - p^{(\sigma)}(t)]^2 \mathrm{d}\lambda_\sigma(t)$ among all polynomials $p \in \mathbb{P}_n$, where $\mathrm{d}\lambda_\sigma$ are given (continuous or discrete) positive measures. The problem can be solved, as in Sect. 2.1.2, by orthogonal polynomials, but they are now orthogonal with respect to the inner product $(u, v)_{H_s} = \sum_{\sigma=0}^{s} \int_{\mathbb{R}} u^{(\sigma)}(t) v^{(\sigma)}(t) \mathrm{d}\lambda_\sigma(t)$ – a so-called Sobolev inner product. This gives rise to Sobolev orthogonal polynomials; see Gautschi [2004, Sect. 1.7] for some history on this problem and relevant literature.

Section 2.1.2. The alternative form (2.25) of computing the coefficients $\hat{c}_j$ was suggested in the 1972 edition of Conte and de Boor [1980] and is further discussed by Shampine [1975]. The Gram–Schmidt procedure described at the end of this section is now called the classical Gram–Schmidt procedure. There are other, modified, versions of Gram–Schmidt that are computationally more effective; see, for example, Björck [1996, pp. 61ff].

Section 2.1.4. The standard text on Fourier series, as already mentioned, is Zygmund [2002], and on orthogonal polynomials, Szegö [1975]. Not only is it true that orthogonal polynomials satisfy a three-term recurrence relation (2.38), but the converse is also true: any system $\{\pi_k\}$ of monic polynomials satisfying (2.38) for all $k \geq 0$, with real coefficients α_k and $\beta_k > 0$, is necessarily orthogonal with respect to some (in general unknown) positive measure. This is known as Favard's Theorem (cf., e.g., Natanson [1965], Vol. 2, Chap. 8, Sect. 6]). The computation of orthogonal polynomials, when the recursion coefficients are not known explicitly, is not an easy task; a number of methods are surveyed in Gautschi [1996]; see also Gautschi [2004, Chap. 2]. Orthogonal systems in $L_2(\mathbb{R})$ that have become prominent in recent years are wavelets, which are functions of the form $\psi_{j,k}(t) = 2^{j/2}\psi(2^j t - k)$, $j, k = 0, \pm 1, \pm 2, \dots$, with ψ a "mother wavelet" – square integrable on $\mathbb{R}$ and (usually) satisfying $\int_{\mathbb{R}} \psi(t)\mathrm{d}t = 0$. Among the growing textbook and monograph literature on this subject, we mention Chui [1992], Daubechies [1992], Walter [1994], Wickerhauser [1994], Hernández and Weiss [1996], Resnikoff and Wells [1998], Burrus et al. [1998], and Novikov et al. [2010].

Section 2.2. Although interpolation by polynomials and spline functions is most common, it is sometimes appropriate to use other systems of approximants for interpolation, for example, trigonometric polynomials or rational functions. Trigonometric interpolation at equally spaced points is closely related to discrete Fourier analysis and hence accessible to the Fast Fourier Transform (FFT). For this, and also for rational interpolation algorithms, see, for example, Stoer and Bulirsch [2002, Sects. 2.1.1 and 2.2]. For the fast Fourier transform and some of its important applications, see Henrici [1979a] and Van Loan [1992].

Besides Lagrange and Hermite interpolation, other types of interpolation processes have been studied in the literature. Among these are Fejér–Hermite interpolation, where one interpolates to given function values and requires the derivative to vanish at these points, and Birkhoff (also called lacunary) interpolation, which is similar to Hermite interpolation, but derivatives of only preselected orders are being interpolated. Remarkably, Fejér–Hermite interpolation at the Chebyshev points (defined in Sect. 2.2.4) converges for every continuous function $f \in C[-1, 1]$. The convergence theory of Lagrange and Fejér–Hermite interpolation is the subject of a monograph by Szabados and Vértesi [1990]. The most comprehensive work on Birkhoff interpolation is the book by G. G. Lorentz et al. [1983]. A more recent monograph by R. A. Lorentz [1992] deals with multivariate Birkhoff interpolation.

Section 2.2.1. The growth of the Lebesgue constants Λ_n is at least $O(\log n)$ as $n \to \infty$; specifically, $\Lambda_n > \frac{2}{\pi}\log n + c$ for any triangular array of interpolation nodes (cf. Sect. 2.1.4), where the constant c can be expressed in terms of Euler's constant γ (cf. Chap. 1, MA 4) by $c = \frac{2}{\pi}\left(\log\frac{8}{\pi} + \gamma\right) = 0.9625228\ldots$; see Rivlin [1990, Theorem 1.2]. The Chebyshev points achieve the optimal order $O(\log n)$; for them, $\Lambda_n \leq \frac{2}{\pi}\log n + 1$ (Rivlin [1990, Theorem 1.2]). Equally spaced nodes, on the other hand, lead to exponential growth of the Lebesgue constants inasmuch as $\Lambda_n \sim 2^{n+1}/(\mathrm{e}n\log n)$ for $n \to \infty$; see Trefethen and Weideman [1991] for some history on this result and Brutman [1997a] for a recent survey on Lebesgue constants. The very last statement of Sect. 2.1.2 is the content of Faber's Theorem (see, e.g., Natanson [1965, Vol. 3, Chap. 2, Theorem 2]), which says that, no matter how one chooses the triangular array of nodes (2.64) in $[a, b]$, there is always a continuous function $f \in C[a, b]$ for which the Lagrange interpolation process does not converge uniformly to f. Indeed, there is an $f \in C[a, b]$ for which Lagrange interpolation diverges almost everywhere in $[a, b]$; see Erdös and Vértesi [1980]. Compare this with Fejér–Hermite interpolation.

Section 2.2.3. A more complete discussion of how the convergence domain of Lagrange interpolation in the complex plane depends on the limit distribution of the interpolation nodes can be found in Krylov [1962, Chap. 12, Sect. 2].

Runge's example is further elucidated in Epperson [1987]. For an analysis of Bernstein's example, we refer to Natanson [1965, Vol. 3, Chap. 2, Sect. 2]. The same divergence phenomenon, incidentally, is exhibited also for a large class of nonequally spaced nodes; see Brutman and Passow [1995]. The proof of Example 5 follows Fejér [1918].

Section 2.2.4. The Chebyshev polynomial arguably is one of the most interesting polynomials from the point of view not only of approximation theory, but also of algebra and number theory. In Rivlin's words, it " ... is like a fine jewel that reveals different characteristics under illumination from various positions." In his text, Rivlin [1990] gives ample testimony in support of this view. Another text, unfortunately available only in Russian (or Polish), is Paszkowski [1983], which has an exhaustive account of analytic properties of Chebyshev polynomials as well as numerical applications.

The convergence result stated in (2.97) follows from (2.59) and the logarithmic growth of Λ_n, since $\mathcal{E}_n(f) \log n \to 0$ for $f \in C^1[-1, 1]$ by Jackson's theorems (cf. Cheney [1998, p. 147]). A more rigorous estimate for the error in (2.102) is $\mathcal{E}_n(f) \le \|\tau_n - f\|_\infty \le \left(4 + \frac{4}{\pi^2} \log n\right) \mathcal{E}_n(f)$ (Rivlin [1990, Theorem 3.3]), where the infinity norm refers to the interval $[-1, 1]$ and $\mathcal{E}_n(f)$ is the best uniform approximation of f on $[-1, 1]$ by polynomials of degree n.

Section 2.2.5. A precursor of the algorithm (2.108) expressing $\lambda_k^{(k)}$ in the form of a sum rather than a product, and thus susceptible to serious cancellation errors, was proposed in Werner [1984]. The more stable algorithm given in the text is due to Berrut and Trefethen [2004]. Barycentric formulae have been developed also for trigonometric interpolation (see Henrici [1979b] for uniform, and Salzer [1949] and Berrut [1984] for nonuniform distributions of the nodes), and for cardinal (sinc-) interpolation (Berrut [1989]); for the latter, see also Gautschi [2001] and Chap. 1, MA 10.

Section 2.2.7. There are explicit formulae, analogous to Lagrange's formula, for Hermite interpolation in the most general case; see, for example, Stoer and Bulirsch [2002, Sect. 2.1.5]. For the important special case $m_k = 2$, see also Chap. 3, Ex. 34(a).

Section 2.2.8. To estimate the error of inverse interpolation, using an appropriate version of (2.60), one needs the derivatives of the inverse function f^{-1}. A general expression for the nth derivative of f^{-1} in terms of the first n derivatives of f is derived in Ostrowski [1973, Appendix C].

Section 2.3. The definition of the class of spline functions $\mathbb{S}_m^k(\Delta)$ can be refined to $\mathbb{S}_m^{\boldsymbol{k}}(\Delta)$, where $\boldsymbol{k}^{\mathrm{T}} = [k_2, k_3, \ldots, k_{n-1}]$ is a vector with integer components $k_i \geq -1$ specifying the degree of smoothness at the interior knots x_i; that is, $s^{(j)}(x_i + 0) - s^{(j)}(x_i - 0) = 0$ for $j = 0, 1, \ldots, k_i$. Then $\mathbb{S}_m^k(\Delta)$ as defined in (2.124) becomes $\mathbb{S}_m^{\boldsymbol{k}}(\Delta)$ with $\boldsymbol{k} = [k, k, \ldots, k]$.

Section 2.3.1. As simple as the procedure of piecewise linear interpolation may seem, it can be applied to advantage in numerical Fourier analysis, for example. In trying to compute the (complex) Fourier coefficients $c_n(f) = \frac{1}{2\pi}\int_0^{2\pi} f(x)\mathrm{e}^{-\mathrm{i}nx}\mathrm{d}x$ of a 2π-periodic function f, one often approximates them by the "discrete Fourier transform" $\hat{c}_n(f) = \frac{1}{N}\sum_{k=0}^{N-1} f(x_k)\mathrm{e}^{-\mathrm{i}nx_k}$, where $x_k = k\frac{2\pi}{N}$. This can be computed efficiently (for large N) by the Fast Fourier Transform. Note, however, that $\hat{c}_n(f)$ is periodic in n with period N, whereas the true Fourier coefficients $c_n(f)$ tend to zero as $n \to \infty$. To remove this deficiency, one can approximate f by some (simple) function φ and thereby approximate $c_n(f)$ by $c_n(\varphi)$. Then $c_n(\varphi)$ will indeed tend to zero as $n \to \infty$. The simplest choice for φ is precisely the piecewise linear interpolant $\varphi = s_1(f;\cdot)$ (relative to the uniform partition of $[0, 2\pi]$ into N subintervals). One then finds, rather remarkably (see Chap. 3, Ex. 14), that $c_n(\varphi)$ is a multiple of the discrete Fourier transform, namely, $c_n(f) = \tau_n\hat{c}_n(f)$, where $\tau_n = \left(\frac{\sin(n\pi/N)}{n\pi/N}\right)^2$; this still allows the application of the FFT but corrects the behavior of $\hat{c}_n(f)$ at infinity. The same modification of the discrete Fourier transform by an "attenuation factor" τ_n occurs for many other approximation processes $f \approx \varphi$; see Gautschi [1971/1972] for a general theory (and history) of attenuation factors.

The near optimality of the piecewise linear interpolant $s_1(f;\cdot)$, as expressed by the inequalities in (2.128), is noted by de Boor [2001, p. 31].

Section 2.3.2. The basis (2.129) for $\mathbb{S}_1^0(\Delta)$ is a special case of a B-spline basis that can be defined for any space of spline functions $\mathbb{S}_m^k(\Delta)$ previously introduced (cf. de Boor [2001, Theorem IX(44)]. The B-splines are formed by means of divided differences of order $m+1$ applied to the truncated power $(t-x)_+^m$ (considered as a function of t). Like the basis in (2.129), each basis function of a B-spline basis is supported on at most $m+1$ consecutive intervals of Δ and is positive on the interior of the support.

Section 2.3.3. A proof of the near optimality of the piecewise linear least squares approximant $\hat{s}_1(f;\cdot)$, as expressed by the inequalities (2.137), can be found in de Boor [2001, p. 32]. For smoothing and least squares approximation procedures involving cubic splines, see, for example, de Boor [2001, Chap. XIV].

Section 2.3.4. (a) For the remark in the last paragraph of (a), see de Boor [2001, Chap. 4, Problem 3].

(b.1) The error bounds in (2.147), which for $r = 0$ and $r = 1$ are asymptotically sharp, are due to Hall and Meyer [1976].

(b.2) The cubic spline interpolant matching second derivatives at the endpoints satisfies the same error bounds as in (2.147) for $r = 0, 1, 2$, with constants $c_0 = \frac{3}{64}$,

$c_1 = \frac{3}{16}$, and $c_2 = \frac{3}{8}$; see Kershaw [1971, Theorem 2]. The same is shown also for periodic spline interpolants s, satisfying $s^{(r)}(a) = s^{(r)}(b)$ for $r = 0, 1, 2$.

(b.3) Even though the natural spline interpolant, in general, converges only with order $|\Delta|^2$ (e.g., for uniform partitions Δ), it has been shown by Atkinson [1968] that the order of convergence is $|\Delta|^4$ on any compact interval contained in the open interval (a, b), and by Kershaw [1971] even on intervals extending (in a sense made precise) to $[a, b]$ as $|\Delta| \to 0$. On such intervals, in fact, the natural spline interpolant s provides approximations to any $f \in C^4[a, b]$ with errors satisfying $\|f^{(r)} - s^{(r)}\|_\infty \leq 8c_r K|\Delta|^{4-r}$, where $K = 2 + \frac{3}{8}\|f^{(4)}\|_\infty$ and $c_0 = \frac{1}{8}$, $c_1 = \frac{1}{2}$, and $c_2 = 1$.

(b.4) The error of the "not-a-knot" spline interpolant is of the same order as the error of the complete spline; it follows from Beatson [1986, (2.49)] that for functions $f \in C^4[a, b]$, one has $\|f^{(r)} - s^{(r)}\|_\infty \leq c_r|\Delta|^{4-r}\|f^{(4)}\|_\infty$, $r = 0, 1, 2$ (at least when $n \geq 6$), where c_r are constants independent of f and Δ. The same bounds are valid for other schemes that depend only on function values, for example, the scheme with m_1 equal to the first (or second) derivative of $p_3(f; x_1, x_2, x_3, x_4; \cdot)$ at $x = a$, and similarly for m_n. The first of these schemes (using first-order derivatives of p_3) is in fact the one recommended by Beatson and Chacko [1989, 1992] for general-purpose interpolation. Numerical experiments in Beatson and Chacko [1989] suggest values of approximately 1 for the constants c_r in the preceding error estimates. In Beatson and Chacko [1992] further comparisons are made among many other cubic spline interpolation schemes.

Section 2.3.5. The minimum norm property of natural splines (Theorem 2.3.1) and its proof based on the identity (2.151), called "the first integral relation" in Ahlberg et al. [1967], is due to Holladay [1957], who derived it in the context of numerical quadrature. "Much of the present-day theory of splines began with this theorem" (Ahlberg et al. [1967, p. 3]). An elegant alternative proof of (2.152), and hence of the theorem, can be based (cf. de Boor [2001, pp. 64–66]) on the Peano representation (see Chap. 3, Sect. 3.2.6) of the second divided difference of $g - s$, that is, $[x_{i-1}, x_i, x_{i+1}](g - s) = \int_{\mathbb{R}} K(t)(g''(t) - s''(t))\mathrm{d}t$, by noting that the Peano kernel K, up to a constant, is the B-spline B_i defined in (2.129). Since the left-hand side is zero by the interpolation properties of g and s, it follows from the preceding equation that $g'' - s''$ is orthogonal to the span of the B_i, hence to s'', which lies in this span.

Exercises and Machine Assignments to Chapter 2

Exercises

1. Suppose you want to approximate the function

$$f(t) = \begin{cases} -1 & \text{if} \quad -1 \leq t < 0, \\ 0 & \text{if} \quad t = 0, \\ 1 & \text{if} \quad 0 < t \leq 1 \end{cases}$$

by a constant function $\varphi(x) = c$:

(a) on [–1,1] in the continuous L_1 norm,
(b) on $\{t_1, t_2, \ldots, t_N\}$ in the discrete L_1 norm,
(c) on [–1,1] in the continuous L_2 norm,
(d) on $\{t_1, t_2, \ldots, t_N\}$ in the discrete L_2 norm,
(e) on [–1,1] in the ∞-norm,
(f) on $\{t_1, t_2, \ldots, t_N\}$ in the discrete ∞-norm.

The weighting in all norms is uniform (i.e., $w(t) \equiv 1$, $w_i = 1$) and $t_i = -1 + \frac{2(i-1)}{N-1}$, $i = 1, 2, \ldots, N$. Determine the best constant c (or constants c, if there is nonuniqueness) and the minimum error.

2. Consider the data

$$f(t_i) = 1, \;\; i = 1, 2, \ldots, N-1; \quad f(t_N) = y \gg 1.$$

(a) Determine the discrete L_∞ approximant to f by means of a constant c (polynomial of degree zero).
(b) Do the same for discrete (equally weighted) least square approximation.
(c) Compare and discuss the results, especially as $N \to \infty$.

3. Let $x_0, x_1, \ldots, x_n$ be pairwise distinct points in $[a, b]$, $-\infty < a < b < \infty$, and $f \in C^1[a, b]$. Show that, given any $\varepsilon > 0$, there exists a polynomial p such that $\|f - p\|_\infty < \varepsilon$ and, at the same time, $p(x_i) = f(x_i)$, $i = 0, 1, \ldots, n$. Here $\|u\|_\infty = \max_{a \le x \le b} |u(x)|$. {*Hint*: write $p = p_n(f; \cdot) + \omega_n q$, where $p_n(f; \cdot)$ is the interpolation polynomial of degree n (cf. Sect. 2.2.1, (2.51)), $\omega_n(x) = \prod_{i=0}^{n}(x - x_i)$, $q \in \mathbb{P}$, and apply Weierstrass's approximation theorem.}
4. Consider the function $f(t) = t^\alpha$ on $0 \le t \le 1$, where $\alpha > 0$. Suppose we want to approximate f best in the L_p norm by a constant c, $0 < c < 1$, that is, minimize the L_p error

$$E_p(c) = \|t^\alpha - c\|_p = \left(\int_0^1 |t^\alpha - c|^p \mathrm{d}t\right)^{1/p}$$

as a function of c. Find the optimal $c = c_p$ for $p = \infty$, $p = 2$, and $p = 1$, and determine $E_p(c_p)$ for each of these p-values.
5. Taylor expansion yields the simple approximation $\mathrm{e}^x \approx 1 + x$, $0 \le x \le 1$. Suppose you want to improve this by seeking an approximation of the form $\mathrm{e}^x \approx 1 + cx$, $0 \le x \le 1$, for some suitable c.

(a) How must c be chosen if the approximation is to be optimal in the (continuous, equally weighted) least squares sense?
(b) Sketch the error curves $e_1(x) := \mathrm{e}^x - (1+x)$ and $e_2(x) := \mathrm{e}^x - (1+cx)$ with c as obtained in (a) and determine $\max_{0 \le x \le 1} |e_1(x)|$ and $\max_{0 \le x \le 1} |e_2(x)|$.
(c) Solve the analogous problem with three instead of two terms in the modified Taylor expansion: $\mathrm{e}^x \approx 1 + c_1 x + c_2 x^2$, and provide error curves for $e_1(x) = \mathrm{e}^x - 1 - x - \frac{1}{2}x^2$ and $e_2(x) = \mathrm{e}^x - 1 - c_1 x - c_2 x^2$.

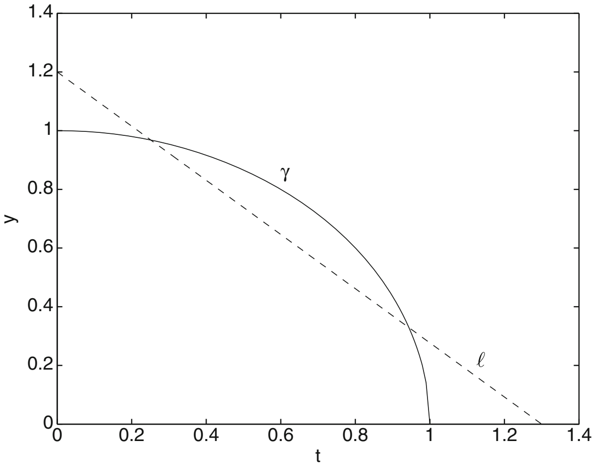
6. Prove Schwarz's inequality

$$|(u, v)| \le \|u\| \cdot \|v\|$$

for the inner product (2.10). {*Hint*: use the nonnegativity of $\|u + tv\|^2$, $t \in \mathbb{R}$.}
7. Discuss uniqueness and nonuniqueness of the least squares approximant to a function f in the case of a discrete set $T = \{t_1, t_2\}$ (i.e., $N = 2$) and $\Phi_n = \mathbb{P}_{n-1}$ (polynomials of degree $\le n-1$). In case of nonuniqueness, determine *all* solutions.
8. Determine the least squares approximation

$$\varphi(t) = \frac{c_1}{1+t} + \frac{c_2}{(1+t)^2}, \quad 0 \le t \le 1,$$

to the exponential function $f(t) = \mathrm{e}^{-t}$, assuming $\mathrm{d}\lambda(t) = \mathrm{d}t$ on [0,1]. Determine the condition number $\mathrm{cond}_\infty \boldsymbol{A} = \|\boldsymbol{A}\|_\infty \|\boldsymbol{A}^{-1}\|_\infty$ of the coefficient matrix $\boldsymbol{A}$ of the normal equations. Calculate the error $f(t) - \varphi(t)$ at $t = 0$, $t = 1/2$, and $t = 1$. {*Point of information*: the integral $\int_1^\infty t^{-m}\mathrm{e}^{-xt}\mathrm{d}t = E_m(x)$ is known as the "mth exponential integral"; cf. Abramowitz and Stegun [1964, (5.1.4)] or Olver et al. [2010, (8.19.3)].}
9. Approximate the circular quarter arc γ given by the equation $y(t) = \sqrt{1-t^2}$, $0 \le t \le 1$ (see figure) by a straight line ℓ in the least squares sense, using either the weight function $w(t) = (1-t^2)^{-1/2}$, $0 \le t \le 1$, or $w(t) = 1$, $0 \le t \le 1$. Where does ℓ intersect the coordinate axes in these two cases? {*Points of information*: $\int_0^{\pi/2} \cos^2\theta \mathrm{d}\theta = \frac{\pi}{4}$, $\int_0^{\pi/2} \cos^3\theta \mathrm{d}\theta = \frac{2}{3}$.}



10. (a) Let the class Φ_n of approximating functions have the following properties. Each $\varphi \in \Phi_n$ is defined on an interval $[a, b]$ symmetric with respect to the

origin (i.e., $a = -b$), and $\varphi(t) \in \Phi_n$ implies $\varphi(-t) \in \Phi_n$. Let $\mathrm{d}\lambda(t) = \omega(t)\mathrm{d}t$, with $\omega(t)$ an even function on $[a,b]$ (i.e., $\omega(-t) = \omega(t)$). Show: if f is an even function on $[a,b]$, then so is its least squares approximant, $\hat{\varphi}_n$, on $[a,b]$ from Φ_n.

(b) Consider the "hat function" $f(t) = \begin{cases} 1-t & \text{if } 0 \le t \le 1, \\ 1+t & \text{if } -1 \le t \le 0. \end{cases}$

Determine its least squares approximation on [–1,1] by a polynomial of degree ≤ 2. (Use $\mathrm{d}\lambda(t) = \mathrm{d}t$.) Simplify your calculation by using part (a). Determine where the error vanishes.

11. Suppose you want to approximate the step function

$$f(t) = \begin{cases} 1 & \text{if } 0 \le t \le 1, \\ 0 & \text{if } t > 1 \end{cases}$$

on the positive line $\mathbb{R}_+$ by a linear combination of exponentials $\pi_j(t) = \mathrm{e}^{-jt}$, $j = 1, 2, \ldots, n$, in the (continuous, equally weighted) least squares sense.

(a) Derive the normal equations. How is the matrix related to the Hilbert matrix?

(b) Use Matlab to solve the normal equations for $n = 1, 2, \ldots, 8$. Print n, the Euclidean condition number of the matrix (supplied by the Matlab function `cond.m`), along with the solution. Plot the approximations vs. the exact function for $1 \le n \le 4$.

12. Let $\pi_j(t) = (t - a_j)^{-1}$, $j = 1, 2, \ldots, n$, where a_j are distinct real numbers with $|a_j| > 1$, $j = 1, 2, \ldots, n$. For $\mathrm{d}\lambda(t) = \mathrm{d}t$ on $-1 \le t \le 1$ and $\mathrm{d}\lambda(t) = 0$, $t \notin [-1, 1]$, determine the matrix of the normal equations for the least squares problem $\int_{\mathbb{R}} (f - \varphi)^2 \mathrm{d}\lambda(t) = \min$, $\varphi = \sum_{j=1}^{n} c_j \pi_j$. Can the sytem $\{\pi_j\}_{j=1}^{n}$, $n > 1$, be an orthogonal system for suitable choices of the constants a_j? Explain.

13. Given an integer $n \ge 1$, consider the subdivision Δ_n of the interval $[0, 1]$ into n equal subintervals of length $1/n$. Let $\pi_j(t)$, $j = 0, 1, \ldots, n$, be the function having the value 1 at $t = j/n$, decreasing on either side linearly to zero at the neighboring subdivision points (if any), and being zero elsewhere.

(a) Draw a picture of these functions. Describe in words the meaning of a linear combination $\pi(t) = \sum_{j=0}^{n} c_j \pi_j(t)$.

(b) Determine $\pi_j(k/n)$ for $j, k = 0, 1, \ldots, n$.

(c) Show that the system $\{\pi_j(t)\}_{j=0}^{n}$ is linearly independent on the interval $0 \le t \le 1$. Is it also linearly independent on the set of subdivision points $0, \frac{1}{n}, \frac{2}{n}, \ldots, \frac{n-1}{n}, 1$ of Δ_n? Explain.

(d) Compute the matrix of the normal equations for $\{\pi_j\}$, assuming $\mathrm{d}\lambda(t) = \mathrm{d}t$ on [0,1]. That is, compute the $(n+1) \times (n+1)$ matrix $\boldsymbol{A} = [a_{ij}]$, where $a_{ij} = \int_0^1 \pi_i(t)\pi_j(t)\mathrm{d}t$.

14. Even though the function $f(t) = \ln(1/t)$ becomes infinite as $t \to 0$, it can be approximated on [0,1] arbitrarily well by polynomials of sufficiently high degree in the (continuous, equally weighted) least squares sense. Show this by proving

$$e_{n,2} := \min_{p \in \mathbb{P}_n} \|f - p\|_2 = \frac{1}{n+1}.$$

{*Hint*: use the following known facts about the "shifted" Legendre polynomial $\pi_j(t)$ of degree j (orthogonal on [0,1] with respect to the weight function $w \equiv 1$ and normalized to satisfy $\pi_j(1) = 1$):

$$\int_0^1 \pi_j^2(t)\mathrm{d}t = \frac{1}{2j+1}, \quad j \geq 0; \quad \int_0^1 \pi_j(t)\ln(1/t)\mathrm{d}t = \begin{cases} 1 & \text{if } j = 0, \\ \frac{(-1)^j}{j(j+1)} & \text{if } j > 0. \end{cases}$$

The first relation is well known from the theory of orthogonal polynomials (see, e.g., Sect. 1.5.1, p. 27 of Gautschi [2004]); the second is due to Blue [1979].}

15. Let $\mathrm{d}\lambda$ be a continuous (positive) measure on $[a, b]$ and $n \geq 1$ a given integer. Assume f continuous on $[a, b]$ and not a polynomial of degree $\leq n-1$. Let $\hat{p}_{n-1} \in \mathbb{P}_{n-1}$ be the least squares approximant to f on $[a, b]$ from polynomials of degree $\leq n-1$:

$$\int_a^b [\hat{p}_{n-1}(t) - f(t)]^2 \mathrm{d}\lambda(t) \leq \int_a^b [p(t) - f(t)]^2 \mathrm{d}\lambda(t), \quad \text{all } p \in \mathbb{P}_{n-1}.$$

Prove: the error $e_n(t) = \hat{p}_{n-1}(t) - f(t)$ changes sign at least n times in $[a, b]$. {*Hint*: assume the contrary and develop a contradiction.}

16. Let f be a given function on [0,1] satisfying $f(0) = 0,\ f(1) = 1$.

    (a) Reduce the problem of approximating f on [0,1] in the (continuous, equally weighted) least squares sense by a quadratic polynomial p satisfying $p(0) = 0,\ p(1) = 1$ to an unconstrained least squares problem (for a different function).

    (b) Apply the result of (a) to $f(t) = t^r,\ r > 2$. Plot the approximation against the exact function for $r = 3$.

17. Suppose you want to approximate $f(t)$ on $[a, b]$ by a function of the form $r(t) = \pi(t)/q(t)$ in the least squares sense with weight function w, where $\pi \in \mathbb{P}_n$ and q is a *given* function (e.g., a polynomial) such that $q(t) > 0$ on $[a, b]$. Formulate this problem as an ordinary polynomial least squares problem for an appropriate new function $\overline{f}$ and new weight function $\overline{w}$.

18. The Bernstein polynomials of degree n are defined by

$$B_j^n(t) = \binom{n}{j} t^j (1-t)^{n-j}, \quad j = 0, 1, \ldots, n,$$

and are usually employed on the interval $0 \leq t \leq 1$.

(a) Show that $B_0^n(0) = 1$, and for $j = 1, 2, \ldots, n$

$$\left.\frac{\mathrm{d}^r}{\mathrm{d}t^r} B_j^n(t)\right|_{t=0} = 0, \quad r = 0, 1, \ldots, j-1; \quad \left.\frac{\mathrm{d}^j}{\mathrm{d}t^j} B_j^n(t)\right|_{t=0} \neq 0.$$

(b) What are the analogous properties at $t = 1$, and how are they most easily derived?

(c) Prepare a plot of the fourth-degree polynomials $B_j^4(t)$, $j = 0, 1, \ldots, 4$, $0 \leq t \leq 1$.

(d) Use (a) to show that the system $\{B_j^n(t)\}_{j=0}^n$ is linearly independent on [0,1] and spans the space $\mathbb{P}_n$.

(e) Show that $\sum_{j=0}^n B_j^n(t) \equiv 1$. {*Hint*: use the binomial theorem.}

19. Prove that, if $\{\pi_j\}_{j=1}^n$ is linearly dependent on the support of $\mathrm{d}\lambda$, then the matrix $\boldsymbol{A} = [a_{ij}]$, where $a_{ij} = (\pi_i, \pi_j)_{\mathrm{d}\lambda} = \int_{\mathbb{R}} \pi_i(t)\pi_j(t)\mathrm{d}\lambda(t)$, is singular.
20. Given the recursion relation $\pi_{k+1}(t) = (t - \alpha_k)\pi_k(t) - \beta_k\pi_{k-1}(t)$, $k = 0, 1, 2, \ldots,$ for the (monic) orthogonal polynomials $\{\pi_j(\,\cdot\,; \mathrm{d}\lambda)\}$, and defining $\beta_0 = \int_{\mathbb{R}} \mathrm{d}\lambda(t)$, show that $\|\pi_k\|^2 = \beta_0\beta_1 \cdots \beta_k$, $k = 0, 1, 2, \ldots.$
21. (a) Derive the three-term recurrence relation

$$\sqrt{\beta_{k+1}}\tilde{\pi}_{k+1}(t) = (t - \alpha_k)\tilde{\pi}_k(t) - \sqrt{\beta_k}\tilde{\pi}_{k-1}, \quad k = 0, 1, 2, \ldots,$$
$$\tilde{\pi}_{-1}(t) = 0, \quad \tilde{\pi}_0 = 1/\sqrt{\beta_0}$$

for the orthonormal polynomials $\tilde{\pi}_k = \pi_k/\|\pi_k\|$, $k = 0, 1, 2, \ldots$.

(b) Use the result of (a) to derive the *Christoffel–Darboux* formula

$$\sum_{k=0}^{n} \tilde{\pi}_k(x)\tilde{\pi}_k(t) = \sqrt{\beta_{n+1}}\, \frac{\tilde{\pi}_{n+1}(x)\tilde{\pi}_n(t) - \tilde{\pi}_n(x)\tilde{\pi}_{n+1}(t)}{x - t}.$$

22. (a) Let $\pi_n(\,\cdot\,) = \pi_n(\,\cdot\,; \mathrm{d}\lambda)$ be the (monic) orthogonal polynomial of degree n relative to the positive measure $\mathrm{d}\lambda$ on $\mathbb{R}$. Show:

$$\int_{\mathbb{R}} \pi_n^2(t; \mathrm{d}\lambda)\mathrm{d}\lambda(t) \leq \int_{\mathbb{R}} p^2(t)\mathrm{d}\lambda(t), \quad \text{all} \;\; p \in \mathring{\mathbb{P}}_n,$$

where $\mathring{\mathbb{P}}_n$ is the class of monic polynomials of degree n. Discuss the case of equality. {*Hint*: represent p in terms of $\pi_j(\,\cdot\,; \mathrm{d}\lambda)$, $j = 0, 1, \ldots, n$.}

(b) If $\mathrm{d}\lambda(t) = \mathrm{d}\lambda_N(t)$ is a discrete measure with exactly N support points $t_1, t_2, \ldots, t_N$, and $\pi_j(t) = \pi_j(\,\cdot\,; \mathrm{d}\lambda_N)$, $j = 0, 1, \ldots, N-1$, are the corresponding (monic) orthogonal polynomials, let $\pi_N(t) = (t - \alpha_{N-1})\pi_{N-1}(t) - \beta_{N-1}\pi_{N-2}(t)$, with α_{N-1}, β_{N-1} defined as in Sect. 2.1.4(2). Show that $\pi_N(t_j) = 0$ for $j = 1, 2, \ldots, N$.

23. Let $\{\pi_j\}_{j=0}^n$ be a system of orthogonal polynomials, not necessarily monic, relative to the (positive) measure $\mathrm{d}\lambda$. For some a_{ij}, define

$$p_i(t) = \sum_{j=0}^{n} a_{ij}\pi_j(t), \quad i = 1, 2, \ldots, n,$$

   (a) Derive conditions on the matrix $\boldsymbol{A} = [a_{ij}]$ which ensure that the system $\{p_i\}_{i=0}^n$ is also a system of orthogonal polynomials.
   (b) Assuming all π_j monic and $\{p_i\}_{i=0}^n$ an orthogonal system, show that each p_i is monic if and only if $\boldsymbol{A} = \boldsymbol{I}$ is the identity matrix.
   (c) Prove the same as in (b), with "monic" replaced by "orthonormal" throughout.

24. Let $(u, v) = \sum_{k=1}^N w_k u(t_k)v(t_k)$ be a discrete inner product on the interval [–1,1] with $-1 \le t_1 < t_2 < \cdots < t_N \le 1$, and let α_k, β_k be the recursion coefficients for the (monic) orthogonal polynomials $\{\pi_k(t)\}_{k=0}^{N-1}$ associated with (u, v):

$$\begin{cases} \pi_{k+1}(t) = (t - \alpha_k)\pi_k(t) - \beta_k\pi_{k-1}(t), \\ \qquad\qquad\qquad\qquad\qquad\qquad k = 0, 1, 2, \ldots, N-2, \\ \pi_0(t) = 1, \ \ \pi_{-1}(t) = 0. \end{cases}$$

   Let $x = \frac{b-a}{2}t + \frac{a+b}{2}$ map the interval [–1,1] to $[a, b]$, and the points $t_k \in [-1, 1]$ to $x_k \in [a, b]$. Define $(u, v)^* = \sum_{k=1}^N w_k u(x_k)v(x_k)$, and let $\{\pi_k^*(x)\}_{k=0}^{N-1}$ be the (monic) orthogonal polynomials associated with $(u, v)^*$. Express the recursion coefficients α_k^*, β_k^* for the $\{\pi_k^*\}$ in terms of those for $\{\pi_k\}$. {*Hint*: first show that $\pi_k^*(x) = (\frac{b-a}{2})^k\pi_k(\frac{2}{b-a}(x - \frac{a+b}{2}))$.}

25. Let

$$(\star) \qquad \begin{cases} \pi_{k+1}(t) = (t - \alpha_k)\pi_k(t) - \beta_k\pi_{k-1}(t), \\ \qquad\qquad\qquad\qquad\qquad\qquad k = 0, 1, 2, \ldots, n-1, \\ \pi_0(t) = 1, \ \ \pi_{-1}(t) = 0 \end{cases}$$

   and consider

$$p_n(t) = \sum_{j=0}^{n} c_j\pi_j(t).$$

   Show that p_n can be computed by the following algorithm (*Clenshaw's algorithm*):

$$(\star\star)\qquad \begin{cases} u_n = c_n, \quad u_{n+1} = 0, \\ u_k = (t-\alpha_k)u_{k+1} - \beta_{k+1}u_{k+2} + c_k, \\ \qquad\qquad\qquad k = n-1, n-2, \dots, 0, \\ p_n = u_0. \end{cases}$$

{*Hint*: write ($\star$) in matrix form in terms of the vector $\boldsymbol{\pi}^{\mathrm{T}} = [\pi_0, \pi_1, \dots, \pi_n]$ and a unit triangular matrix. Do likewise for ($\star\star$).}

26. Show that the elementary Lagrange interpolation polynomials $\ell_i(x)$ are invariant with respect to any linear transformation of the independent variable.
27. Use Matlab to prepare plots of the Lebesgue function for interpolation, $\lambda_n(x)$, $-1 \le x \le 1$, for $n = 5, 10, 20$, with the interpolation nodes x_i being given by

    (a) $x_i = -1 + \frac{2i}{n}, \quad i = 0, 1, 2, \dots, n$;
    (b) $x_i = \cos \frac{2i+1}{2n+2}\pi, \quad i = 0, 1, 2, \dots, n$.

    Compute $\lambda_n(x)$ on a grid obtained by dividing each interval $[x_{i-1}, x_i]$, $i = 1, 2, \dots, n$, into 20 equal subintervals. Plot $\log_{10} \lambda_n(x)$ in case (a), and $\lambda_n(x)$ in case (b). Comment on the results.
28. Let $\omega_n(x) = \prod_{k=0}^{n} (x-k)$ and denote by x_n the location of the extremum of ω_n on [0,1], that is, the unique x in [0,1], where $\omega_n'(x) = 0$.

    (a) Prove or disprove that $x_n \to 0$ as $n \to \infty$.
    (b) Investigate the monotonicity of x_n as n increases.

29. Consider equidistant sampling points $x_k = k$ ($k = 0, 1, \dots, n$) and $\omega_n(x) = \prod_{k=0}^{n} (x-k)$, $0 \le x \le n$.

    (a) Show that $\omega_n(x) = (-1)^{n+1}\omega_n(n-x)$. What kind of symmetry does this imply?
    (b) Show that $|\,\omega_n(x)| < |\,\omega_n(x+1)|$ for nonintegral $x > (n-1)/2$.
    (c) Show that the relative maxima of $|\,\omega_n(x)|$ increase monotonically (from the center of $[0, n]$ outward).

30. Let

$$\lambda_n(x) = \sum_{i=0}^{n} |\ell_i(x)|$$

be the Lebesgue function for polynomial interpolation at the distinct points $x_i \in [a, b]$, $i = 0, 1, \dots, n$, and $\Lambda_n = \|\lambda_n\|_\infty = \max_{a \le x \le b} |\lambda_n(x)|$ the Lebesgue constant. Let $p_n(f; \cdot)$ be the polynomial of degree $\le n$ interpolating f at the nodes x_i. Show that in the inequality

$$\|p_n(f; \cdot)\|_\infty \le \Lambda_n \|f\|_\infty, \quad f \in C[a, b],$$

equality can be attained for some $f = \varphi \in C[a,b]$. {*Hint*: let $\|\lambda_n\|_\infty = \lambda_n(x_\infty)$; take $\varphi \in C[a,b]$ piecewise linear and such that $\varphi(x_i) = \operatorname{sgn} \ell_i(x_\infty)$, $i = 0, 1, \ldots, n$.}

31. (a) Let $x_0, x_1, \ldots, x_n$ be $n+1$ distinct points in $[a,b]$ and $f_i = f(x_i)$, $i = 0, 1, \ldots, n$, for some function f. Let $f_i^* = f_i + \varepsilon_i$, where $|\varepsilon_i| \le \varepsilon$. Use the Lagrange interpolation formula to show that $|p_n(f^*; x) - p_n(f; x)| \le \varepsilon\lambda_n(x)$, $a \le x \le b$, where $\lambda_n(x)$ is the Lebesgue function (cf. Ex. 30).
    (b) Show: $\lambda_n(x_j) = 1$ for $j = 0, 1, \ldots, n$.
    (c) For quadratic interpolation at three equally spaced points, show that $\lambda_2(x) \le 1.25$ for any x between the three points.
    (d) Obtain $\lambda_2(x)$ for $x_0 = 0$, $x_1 = 1$, $x_2 = p$, where $p \gg 1$, and determine $\max_{1 \le x \le p} \lambda_2(x)$. How fast does this maximum grow with p? {*Hint*: to simplify the algebra, note from (b) that $\lambda_2(x)$ on $1 \le x \le p$ must be of the form $\lambda_2(x) = 1 + c(x-1)(p-x)$ for some constant c.}
32. In a table of the Bessel function $J_0(x) = \frac{1}{\pi}\int_0^\pi \cos(x \sin\theta)\mathrm{d}\theta$, where x is incremented in steps of size h, how small must h be chosen if the table is to be "linearly interpolable" with error less that 10^{-6} in absolute value? {*Point of information*: $\int_0^{\pi/2} \sin^2\theta \mathrm{d}\theta = \frac{\pi}{4}$.}
33. Suppose you have a table of the logarithm function $\ln x$ for positive integer values of x, and you compute $\ln 11.1$ by quadratic interpolation at $x_0 = 10$, $x_1 = 11$, $x_2 = 12$. Estimate the relative error incurred.
34. The "Airy function" $y(x) = \mathrm{Ai}(x)$ is a solution of the differential equation $y'' = xy$ satisfying appropriate initial conditions. It is known that $\mathrm{Ai}(x)$ on $[0, \infty)$ is monotonically decreasing to zero and $\mathrm{Ai}'(x)$ monotonically increasing to zero. Suppose you have a table of Ai and Ai$'$ (with tabular step h) and you want to interpolate

    (a) linearly between x_0 and x_1,
    (b) quadratically between x_0, x_1, and x_2,

    where x_0, $x_1 = x_0 + h$, $x_2 = x_0 + 2h$ are (positive) tabular arguments. Determine close upper bounds for the respective errors in terms of quantities $y_k = y(x_k)$, $y_k' = y'(x_k)$, $k = 0, 1, 2$, contained in the table.
35. The error in linear interpolation of f at x_0, x_1 is known to be

$$f(x) - p_1(f; x) = (x - x_0)(x - x_1)\,\frac{f''(\xi(x))}{2}, \quad x_0 < x < x_1,$$

    if $f \in C^2[x_0, x_1]$. Determine $\xi(x)$ explicitly in the case $f(x) = \frac{1}{x}$, $x_0 = 1$, $x_1 = 2$, and find $\max_{1 \le x \le 2} \xi(x)$ and $\min_{1 \le x \le 2} \xi(x)$.
36. (a) Let $p_n(f; x)$ be the interpolation polynomial of degree $\le n$ interpolating $f(x) = \mathrm{e}^x$ at the points $x_i = i/n$, $i = 0, 1, 2, \ldots, n$. Derive an upper bound for

$$\max_{0 \le x \le 1} |\mathrm{e}^x - p_n(f; x)|$$

and determine the smallest n guaranteeing an error less than 10^{-6} on [0, 1]. {*Hint*: first show that for any integer i with $0 \le i \le n$ one has $\max_{0\le x\le 1} |(x-\frac{i}{n})(x-\frac{n-i}{n})| \le \frac{1}{4}$.}

(b) Solve the analogous problem for the nth-degree Taylor polynomial $t_n(x) = 1 + x + \frac{x^2}{2!} + \cdots + \frac{x^n}{n!}$, and compare the result with the one in (a).

37. Let $x_0 < x_1 < x_2 < \cdots < x_n$ and $H = \max_{0\le i\le n-1}(x_{i+1} - x_i)$. Defining $\omega_n(x) = \prod_{i=0}^{n}(x-x_i)$, find an upper bound for $\|\omega_n\|_\infty = \max_{x_0\le x\le x_n} |\omega_n(x)|$ in terms of H and n. {*Hint*: assume $x_j \le x \le x_{j+1}$ for some $0 \le j < n$ and estimate $(x - x_j)(x - x_{j+1})$ and $\prod_{\substack{i\ne j\\ i\ne j+1}} (x - x_i)$ separately.}

38. Show that the power x^n on the interval $-1 \le x \le 1$ can be uniformly approximated by a linear combination of powers $1, x, x^2, \ldots, x^{n-1}$ with error $\le 2^{-(n-1)}$. In this sense, the powers of x become "less and less linearly independent" on [–1,1] with growing exponent n.

39. Determine

$$\min \max_{a\le x\le b} |a_0x^n + a_1x^{n-1} + \cdots + a_n|, \quad n \ge 1,$$

where the minimum is taken over all real $a_0, a_1, \ldots, a_n$ with $a_0 \ne 0$. {*Hint*: use Theorem 2.2.1.}

40. Let $a > 1$ and $\mathbb{P}_n^a = \{p \in \mathbb{P}_n : p(a) = 1\}$. Define $\hat{p}_n \in \mathbb{P}_n^a$ by $\hat{p}_n(x) = \frac{T_n(x)}{T_n(a)}$, where T_n is the Chebyshev polynomial of degree n, and let $\|\cdot\|_\infty$ denote the maximum norm on the interval $[-1, 1]$. Prove:

$$\|\hat{p}_n\|_\infty \le \|p\|_\infty \quad \text{for all} \quad p \in \mathbb{P}_n^a.$$

{*Hint*: imitate the proof of Theorem 2.2.1.}

41. Let

$$f(x) = \int_5^\infty \frac{\mathrm{e}^{-t}}{t-x}\mathrm{d}t, \qquad -1 \le x \le 1,$$

and let $p_{n-1}(f; \cdot)$ be the polynomial of degree $\le n-1$ interpolating f at the n Chebyshev points $x_\nu = \cos(\frac{2\nu-1}{2n}\pi)$, $\nu = 1, 2, \ldots, n$. Derive an upper bound for $\max_{-1\le x\le 1} |f(x) - p_{n-1}(f, x)|$.

42. Let f be a positive function defined on $[a, b]$ and assume

$$\min_{a\le x\le b} |f(x)| = m_0, \quad \max_{a\le x\le b} |f^{(k)}(x)| = M_k, \ \ k = 0, 1, 2, \ldots.$$

(a) Denote by $p_{n-1}(f; \cdot)$ the polynomial of degree $\le n-1$ interpolating f at the n Chebyshev points (relative to the interval $[a, b]$). Estimate the maximum relative error $r_n = \max_{a\le x\le b} |(f(x) - p_{n-1}(f; x))/f(x)|$.

(b) Apply the result of (a) to $f(x) = \ln x$ on $I_r = \{\mathrm{e}^r \le x \le \mathrm{e}^{r+1}\}$, $r \ge 1$ an integer. In particular, show that $r_n \le \alpha(r, n)c^n$, where $0 < c < 1$ and α is slowly varying. Exhibit c.

(c) (This relates to the function $f(x) = \ln x$ of part (b).) How does one compute $f(\overline{x})$, $\overline{x} \in I_s$, from $f(x)$, $x \in I_r$?

43. (a) For quadratic interpolation on equally spaced points x_0, $x_1 = x_0 + h$, $x_2 = x_0 + 2h$, derive an upper bound for $\|f - p_2(f;\cdot)\|_\infty$ involving $\|f'''\|_\infty$ and h. (Here $\|u\|_\infty = \max_{x_0 \le x \le x_2} |u(x)|$.)
(b) Compare the bound obtained in (a) with the analogous one for interpolation at the three Chebyshev points on $[x_0, x_2]$.
44. (a) Suppose the function $f(x) = \ln(2 + x)$, $-1 \le x \le 1$, is interpolated by a polynomial p_n of degree $\le n$ at the Chebyshev points $x_k = \cos\left(\frac{2k+1}{2n+2}\pi\right)$, $k = 0, 1, \dots, n$. Derive a bound for the maximum error $\|f - p_n\|_\infty = \max_{-1 \le x \le 1} |f(x) - p_n(x)|$.
(b) Compare the result of (a) with bounds for $\|f - t_n\|_\infty$, where $t_n(x)$ is the nth-degree Taylor polynomial of f and where either Lagrange's form of the remainder is used or the full Taylor expansion of f.
45. Consider $f(t) = \cos^{-1} t$, $-1 \le t \le 1$. Obtain the least squares approximation $\hat{\varphi}_n \in \mathbb{P}_n$ of f relative to the weight function $w(t) = (1-t)^{-\frac{1}{2}}$; that is, find the solution $\varphi = \hat{\varphi}_n$ of

$$\text{minimize} \quad \left\{ \int_{-1}^{1} [f(t) - \varphi(t)]^2 \frac{\mathrm{d}t}{\sqrt{1-t^2}} \ : \ \varphi \in \mathbb{P}_n \right\}.$$

Express $\hat{\varphi}_n$ in terms of Chebyshev polynomials $\pi_j(t) = T_j(t)$.
46. Compute $T_n'(0)$, where T_n is the Chebyshev polynomial of degree n.
47. Prove that the system of Chebyshev polynomials $\{T_k : 0 \le k < n\}$ is orthogonal with respect to the discrete inner product $(u, v) = \sum_{\nu=1}^{n} u(x_\nu)v(x_\nu)$, where x_ν are the Chebyshev points $x_\nu = \cos \frac{2\nu-1}{2n}\pi$.
48. Let $T_k(x)$ denote the Chebyshev polynomial of degree k. Clearly, $T_n(T_m(x))$ is a polynomial of degree $n \cdot m$. Identify it.
49. Let T_n denote the Chebyshev polynomial of degree $n \ge 2$. The equation

$$x = T_n(x)$$

is an algebraic equation of degree n and hence has exactly n roots. Identify them.
50. For any x with $0 \le x \le 1$ show that $T_n(2x - 1) = T_{2n}(\sqrt{x})$.
51. Let $f(x)$ be defined for all $x \in \mathbb{R}$ and infinitely often differentiable on $\mathbb{R}$. Assume further that

$$|f^{(m)}(x)| \le 1, \quad \text{all } x \in \mathbb{R}, \ m = 1, 2, 3, \dots.$$

Let $h > 0$ and p_{2n-1} be the polynomial of degree $< 2n$ interpolating f at the $2n$ points $x = kh$, $k = \pm 1, \pm 2, \dots, \pm n$. For what values of h is it true that

$$\lim_{n \to \infty} p_{2n-1}(0) = f(0)\,?$$

(Note that $x = 0$ is *not* an interpolation node.) Explain why the convergence theory discussed in Sect. 2.2.3 does not apply here. {*Point of information*: $n! \sim \sqrt{2\pi n}(n/\mathrm{e})^n$ as $n \to \infty$ (Stirling's formula).}

52. (a) Let $x_i^C = \cos\left(\frac{2i+1}{2n+2}\pi\right)$, $i = 0, 1, \ldots, n$, be Chebyshev points on $[-1, 1]$. Obtain the analogous Chebyshev points t_i^C on $[a, b]$ (where $a < b$) and find an upper bound of $\prod_{i=0}^{n}(t - t_i^C)$ for $a \le t \le b$.
    (b) Consider $f(t) = \ln t$ on $[a, b]$, $0 < a < b$, and let $p_n(t) = p_n(f; t_0^{(n)}, t_1^{(n)}, \ldots, t_n^{(n)}; t)$. Given $a > 0$, how large can b be chosen such that $\lim_{n\to\infty} p_n(t) = f(t)$ for arbitrary nodes $t_i^{(n)} \in [a, b]$ and arbitrary $t \in [a, b]$?
    (c) Repeat (b), but with $t_i^{(n)} = t_i^C$ (see (a)).
53. Let $\mathbb{P}_m^+$ be the set of all polynomials of degree $\le m$ that are nonnegative on the real line,

$$\mathbb{P}_m^+ = \{p : \quad p \in \mathbb{P}_m,\ p(x) \ge 0 \text{ for all } x \in \mathbb{R}\}.$$

    Consider the following interpolation problem: find $p \in \mathbb{P}_m^+$ such that $p(x_i) = f_i$, $i = 0, 1, \ldots, n$, where $f_i \ge 0$ and x_i are distinct points on $\mathbb{R}$.

    (a) Show that, if $m = 2n$, the problem admits a solution for arbitrary $f_i \ge 0$.
    (b) Prove: if a solution is to exist for arbitrary $f_i \ge 0$, then, necessarily, $m \ge 2n$. {*Hint*: consider $f_0 = 1,\ f_1 = f_2 = \cdots = f_n = 0$.}

54. Defining forward differences by $\Delta f(x) = f(x + h) - f(x)$, $\Delta^2 f(x) = \Delta(\Delta f(x)) = f(x + 2h) - 2f(x + h) + f(x)$, and so on, show that

$$\Delta^k f(x) = k!h^k [x_0, x_1, \ldots, x_k]f,$$

    where $x_j = x + jh,\ j = 0, 1, 2, \ldots$. Prove an analogous formula for backward differences.
55. Let $f(x) = x^7$. Compute the fifth divided difference $[0,1,1,1,2,2]f$ of f. It is known that this divided difference is expressible in terms of the fifth derivative of f evaluated at some ξ, $0 < \xi < 2$ (cf. (2.117)). Determine ξ.
56. In this problem $f(x) = \mathrm{e}^x$ throughout.

    (a) Prove: for any real number t, one has

$$[t, t + 1, \ldots, t + n]f = \frac{(\mathrm{e} - 1)^n}{n!}\mathrm{e}^t.$$

    {*Hint*: use induction on n.}
    (b) From (2.117) we know that

$$[0, 1, \ldots, n]f = \frac{f^{(n)}(\xi)}{n!}, \quad 0 < \xi < n.$$

    Use the result in (a) to determine ξ. Is ξ located to the left or to the right of the midpoint $n/2$?

57. (Euler, 1734) Let $x_k = 10^k$, $k = 0, 1, 2, 3, \ldots$, and $f(x) = \log_{10} x$.
   (a) Show that

$$[x_0, x_1, \ldots, x_n]f = \frac{(-1)^{n-1}}{10^{n(n-1)/2}(10^n - 1)}, \quad n = 1, 2, 3, \ldots.$$

   {*Hint*: prove more generally

$$[x_r, x_{r+1}, \ldots, x_{r+n}]f = \frac{(-1)^{n-1}}{10^{rn+n(n-1)/2}(10^n - 1)}, \quad r \geq 0,$$

   by induction on n.}
   (b) Use Newton's interpolation formula to determine $p_n(x) = p_n(f; x_0, x_1, \ldots, x_n; x)$. Show that $\lim_{n\to\infty} p_n(x)$ exists for $1 \leq x < 10$. Is the limit equal to $\log_{10} x$? (Check, e.g., for $x = 9$.)

58. Show that

$$\frac{\partial}{\partial x_0}[x_0, x_1, \ldots, x_n]f = [x_0, x_0, x_1, \ldots, x_n]f,$$

   assuming f is differentiable at x_0. What about the partial derivative with respect to one of the other variables?

59. (a) For $n + 1$ distinct nodes x_ν, show that

$$[x_0, x_1, \ldots, x_n]f = \sum_{\nu=0}^{n} \frac{f(x_\nu)}{\prod_{\mu\neq\nu}(x_\nu - x_\mu)}.$$

   (b) Show that

$$[x_0, x_1, \ldots, x_n](fg_j) = [x_0, x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n]f,$$

   where $g_j(x) = x - x_j$.

60. (Mikeladze, 1941) Assuming $x_0, x_1, \ldots, x_n$ mutually distinct, show that

$$[\underbrace{x_0, x_0, \ldots, x_0}_{m \text{ times}}, x_1, x_2, \ldots, x_n]f$$

$$= \frac{[\overbrace{x_0, \ldots, x_0}^{m \text{ times}}]f}{\prod_{\mu=1}^{n}(x_0 - x_\mu)} + \sum_{\nu=1}^{n} \frac{[\overbrace{x_0, \ldots, x_0}^{(m-1) \text{ times}}, x_\nu]f}{\prod_{\substack{\mu=0\\ \mu\neq\nu}}^{n}(x_\nu - x_\mu)}.$$

   {*Hint*: use induction on m.}

61. Determine the number of additions and the number of multiplications/divisions required

   (a) to compute all divided differences for $n + 1$ data points,
   (b) to compute all auxiliary quantities $\lambda_i^{(n)}$ in (2.103), and

(c) to compute $p_n(f;\cdot)$ (efficiently) from Newton's formula (2.111), once the divided differences are available. Compare with the analogous count for the barycentric formula (2.105), assuming all auxiliary quantities available. Overall, which, if any, of the two formulae can be computed more economically?

62. Consider the data $f(0) = 5$, $f(1) = 3$, $f(3) = 5$, $f(4) = 12$.

(a) Obtain the appropriate interpolation polynomial $p_3(f;x)$ in Newton's form.
(b) The data suggest that f has a minimum between $x = 1$ and $x = 3$. Find an approximate value for the location $x_{\min}$ of the minimum.

63. Let $f(x) = (1+a)^x$, $|a| < 1$. Show that $p_n(f;0,1,\dots,n;x)$ is the truncation of the binomial series for f to $n+1$ terms. {*Hint*: use Newton's form of the interpolation polynomial.}
64. Suppose f is a function on [0,3] for which one knows that

$$f(0) = 1,\quad f(1) = 2,\quad f'(1) = -1,\quad f(3) = f'(3) = 0.$$

(a) Estimate $f(2)$, using Hermite interpolation.
(b) Estimate the maximum possible error of the answer given in (a) if one knows, in addition, that $f \in C^5[0,3]$ and $|f^{(5)}(x)| \le M$ on [0,3]. Express the answer in terms of M.

65. (a) Use Hermite interpolation to find a polynomial of lowest degree satisfying $p(-1) = p'(-1) = 0$, $p(0) = 1$, $p(1) = p'(1) = 0$. Simplify your expression for p as much as possible.
(b) Suppose the polynomial p of (a) is used to approximate the function $f(x) = [\cos(\pi x/2)]^2$ on $-1 \le x \le 1$.

(b1) Express the error $e(x) = f(x) - p(x)$ (for some *fixed* x in $[-1,1]$) in terms of an appropriate derivative of f.
(b2) Find an upper bound for $|e(x)|$ (still for a *fixed* $x \in [-1,1]$).
(b3) Estimate $\max_{-1\le x\le 1} |e(x)|$.

66. Consider the problem of finding a polynomial $p \in \mathbb{P}_n$ such that

$$p(x_0) = f_0,\quad p'(x_i) = f_i',\quad i = 1,2,\dots,n,$$

where x_i, $i = 1,2,\dots,n$, are distinct nodes. (It is not excluded that $x_1 = x_0$.) This is neither a Lagrange nor a Hermite interpolation problem (why not?). Nevertheless, show that the problem has a unique solution and describe how it can be obtained.
67. Let

$$f(t) = \begin{cases} 0 & \text{if} \quad 0 \le t \le \frac{1}{2}, \\ 1 & \text{if} \quad \frac{1}{2} \le t \le 1. \end{cases}$$

(a) Find the linear least squares approximant $\hat{p}_1$ to f on $[0,1]$, that is, the polynomial $p_1 \in \mathbb{P}_1$ for which

$$\int_0^1 [p_1(t) - f(t)]^2 \mathrm{d}t = \min.$$

Use the normal equations with $\pi_0(t) = 1,\ \pi_1(t) = t$.

(b) Can you do better with continuous piecewise linear functions (relative to the partition $[0,1] = [0,\frac{1}{2}] \cup [\frac{1}{2},1]$)? Use the normal equations for the B-spline basis $B_0,\ B_1,\ B_2$ (cf. Sect. 2.2.2 and Ex. 13).

68. Show that $\mathbb{S}_m^m(\Delta) = \mathbb{P}_m$.
69. Let Δ be the subdivision

$$\Delta = [0,1] \cup [1,2] \cup [2,3]$$

of the interval [0,3]. Define the function s by

$$s(x) = \begin{cases} 2 - x(3 - 3x + x^2) & \text{if} \quad 0 \le x \le 1, \\ 1 & \text{if} \quad 1 \le x \le 2, \\ \frac{1}{4}\, x^2(3 - x) & \text{if} \quad 2 \le x \le 3. \end{cases}$$

To which class $\mathbb{S}_m^k(\Delta)$ does s belong?

70. In

$$s(x) = \begin{cases} p(x) & \text{if} \ \ 0 \le x \le 1, \\ (2-x)^3 & \text{if} \ \ 1 \le x \le 2 \end{cases}$$

determine $p \in \mathbb{P}_3$ such that $s(0) = 0$ and s is a cubic spline in $\mathbb{S}_3^2(\Delta)$ on the subdivision $\Delta = [0,1] \cup [1,2]$ of the interval [0,2]. Do you get a natural spline?

71. Let Δ: $a = x_1 < x_2 < x_3 < \cdots < x_n = b$ be a subdivision of $[a,b]$ into $n-1$ subintervals. What is the dimension of the space $\mathbb{S}_m^k = \{s \in C^k[a,b]: s|_{[x_i,x_{i+1}]} \in \mathbb{P}_m,\ i = 1,2,\ldots,n-1\}$?
72. Given the subdivision $\Delta:\ a = x_1 < x_2 < \cdots < x_n = b$ of $[a,b]$, determine a basis of "hat functions" for the space $\boldsymbol{S} = \{s \in \mathbb{S}_1^0 :\ s(a) = s(b) = 0\}$.
73. Let $\Delta\ :\ a = x_1 < x_2 < x_3 < \cdots < x_{n-1} < x_n = b$ be a subdivision of $[a,b]$ into $n-1$ subintervals. Suppose we are given values $f_i = f(x_i)$ of some function $f(x)$ at the points $x = x_i$, $i = 1,2,\ldots,n$. In this problem $s \in \mathbb{S}_2^1$ is a quadratic spline in $C^1[a,b]$ that interpolates f on Δ, that is, $s(x_i) = f_i$, $i = 1,2,\ldots,n$.

(a) Explain why one expects an additional condition to be required in order to determine s uniquely.

(b) Define $m_i = s'(x_i)$, $i = 1,2,\ldots,n-1$. Determine $p_i := s\big|_{[x_i,x_{i+1}]}$, $i = 1,2,\ldots,n-1$, in terms of f_i, f_{i+1}, and m_i.

(c) Suppose one takes $m_1 = f'(a)$. (According to (a), this determines s uniquely.) Show how $m_2, m_3, \ldots, m_{n-1}$ can be computed.

74. Let the subdivision Δ of $[a, b]$ be given by

$$\Delta: \; a = x_1 < x_2 < x_3 < \cdots < x_{n-1} < x_n = b, \; n \geq 2,$$

and let $f_i = f(x_i)$, $i = 1, 2, \ldots, n$, for some function f. Suppose you want to interpolate this data by a quintic spline $s_5(f; \cdot)$ (a piecewise fifth-degree polynomial of smoothness class $C^4[a, b]$). By counting the number of parameters at your disposal and the number of conditions imposed, state how many additional conditions (if any) you expect are needed to make $s_5(f; \cdot)$ unique.

75. Let

$$\Delta: \quad a = x_1 < x_2 < x_3 < \cdots < x_{n-1} < x_n = b.$$

Consider the following problem: given $n-1$ numbers f_ν and $n-1$ points ξ_ν with $x_\nu < \xi_\nu < x_{\nu+1}$ $(\nu = 1, 2, \ldots, n-1)$, find a piecewise linear function $s \in \mathbb{S}_1^0(\Delta)$ such that

$$s(\xi_\nu) = f_\nu \quad (\nu = 1, 2, \ldots, n-1), \quad s(x_1) = s(x_n).$$

Representing s in terms of the basis $B_1, B_2, \ldots, B_n$ of "hat functions," determine the structure of the linear system of equations that you obtain for the coefficients c_j in $s(x) = \sum_{j=1}^n c_j B_j(x)$. Describe how you would solve the system.

76. Let $s_1(x) = 1 + c(x+1)^3$, $-1 \leq x \leq 0$, where c is a (real) parameter. Determine $s_2(x)$ on $0 \leq x \leq 1$ so that

$$s(x) := \begin{cases} s_1(x) & \text{if} \quad -1 \leq x \leq 0, \\ s_2(x) & \text{if} \quad 0 \leq x \leq 1 \end{cases}$$

is a natural cubic spline on $[-1, 1]$ with knots at $-1, 0, 1$. How must c be chosen if one wants $s(1) = -1$?

77. Derive (2.136).

78. Determine the quantities m_i in the variant of piecewise cubic Hermite interpolation mentioned at the end of Sect. 2.3.4(a).

79. (a) Derive the two extra equations for $m_1, m_2, \ldots, m_n$ that result from the "not-a-knot" condition (Sect. 2.3.4, (b.4)) imposed on the cubic spline interpolant $s \in \mathbb{S}_3^2(\Delta)$ (with Δ as in Ex. 73).

(b) Adjoin the first of these equations to the top and the second to the bottom of the system of $n-2$ equations derived in Sect. 2.3.4(b). Then apply elementary row operations to produce a tridiagonal system. Display the new matrix elements in the first and last equations, simplified as much as possible.

(c) Is the tridiagonal system so obtained diagonally dominant?

80. Let $\mathbb{S}_1^0(\Delta)$ be the class of continuous piecewise linear functions relative to the subdivision $a = x_1 < x_2 < \cdots < x_n = b$. Let $\|g\|_\infty = \max_{a \le x \le b} |g(x)|$, and denote by $s_1(g;\,\cdot)$ the piecewise linear interpolant (from $\mathbb{S}_1^0(\Delta)$) to g.

    (a) Show: $\|s_1(g;\,\cdot)\|_\infty \le \|g\|_\infty$ for any $g \in C[a,b]$.
    (b) Show: $\|f - s_1(f;\,\cdot)\|_\infty \le 2\|f - s\|_\infty$ for any $s \in \mathbb{S}_1^0$, $f \in C[a,b]$. {*Hint*: use additivity of $s_1(f;\,\cdot)$ with respect to f.}
    (c) Interpret the result in (b) when s is the best uniform spline approximant to f.

81. Consider the interval $[a,b] = [-1,1]$ and its subdivision $\Delta = [-1,0] \cup [0,1]$, and let $f(x) = \cos \frac{\pi}{2} x$, $-1 \le x \le 1$.

    (a) Determine the natural cubic spline interpolant to f on Δ.
    (b) Illustrate Theorem 2.3.2 by taking in turn $g(x) = p_2(f; -1, 0, 1; x)$ and $g(x) = f(x)$.
    (c) Discuss analogously the complete cubic spline interpolant to f on Δ' (cf. (2.149)) and the choices $g(x) = p_3(f; -1, 0, 1, 1; x)$ and $g(x) = f(x)$.

Machine Assignments

1. (a) A simple-minded approach to best uniform approximation of a function $f(x)$ on [0,1] by a linear function $ax + b$ is to first discretize the problem and then, for various (appropriate) trial values of a, solve the problem of (discrete) uniform approximation of $f(x) - ax$ by a constant b (which admits an easy solution). Write a program to implement this idea.
    (b) Run your program for $f(x) = \mathrm{e}^x$, $f(x) = 1/(1+x)$, $f(x) = \sin \frac{\pi}{2} x$, $f(x) = x^\alpha$ ($\alpha = 2,3,4,5$). Print the respective optimal values of a and b and the associated minimum error. What do you find particularly interesting in the results (if anything)?
    (c) Give a heuristic explanation (and hence exact values) for the results, using the known fact that the error curve for the optimal linear approximation attains its maximum modulus at three consecutive points $0 \le x_0 < x_1 < x_2 \le 1$ with alternating signs (*Principle of Alternation*).

2. (a) Determine the $(n+1) \times (n+1)$ matrix $\boldsymbol{A} = [a_{ij}]$, $a_{ij} = (B_i^n, B_j^n)$, of the normal equations relative to the Bernstein basis

$$B_j^n(t) = \binom{n}{j} t^j (1-t)^{n-j}, \quad j = 0, 1, \ldots, n,$$

    and weight function $w(t) \equiv 1$ on [0,1]. {*Point of information*: $\int_0^1 t^k (1-t)^\ell \mathrm{d}t = k!\ell!/(k+\ell+1)!$}.

(b) Use Matlab to solve the normal equations of (a) for $n = 5 : 5 : 25$, when the function to be approximated is $f(t) \equiv 1$. What should the exact answer be? For each n, print the infinity norm of the error vector and an estimate of the condition number of $\boldsymbol{A}$. Comment on your results.

3. Compute discrete least squares approximations to the function $f(t) = \sin\left(\frac{\pi}{2}t\right)$ on $0 \le t \le 1$ by polynomials of the form

$$\varphi_n(t) = t + t(1-t)\sum_{j=1}^{n} c_j t^{j-1},\ n = 1(1)5,$$

using N abscissae $t_k = k/(N+1)$, $k = 1, 2, \ldots, N$, and equal weights 1. Note that $\varphi_n(0) = 0$, $\varphi_n(1) = 1$ are the exact values of f at $t = 0$ and $t = 1$, respectively. $\{$*Hint*: approximate $f(t) - t$ by a linear combination of $\pi_j(t) = t^j(1-t)$; $j = 1, 2, \ldots, n$.$\}$ Write a Matlab program for solving the normal equations $\boldsymbol{A}\boldsymbol{c} = \boldsymbol{b}$, $\boldsymbol{A} = [(\pi_i, \pi_j)]$, $\boldsymbol{b} = [(\pi_i, f - t)]$, $\boldsymbol{c} = [c_j]$, that does the computation in both single and double precision. For each $n = 1, 2, \ldots, 5$ output the following:

- the condition number of the system (computed in double precision);
- the maximum relative error in the coefficients, $\max_{1\le j\le n} |(c_j^s - c_j^d)/c_j^d|$, where c_j^s are the single-precision values of c_j and c_j^d the double-precision values;
- the minimum and maximum error (computed in double precision),

$$e_{\min} = \min_{1\le k\le N} |\varphi_n(t_k) - f(t_k)|,\ \ e_{\max} = \max_{1\le k\le N} |\varphi_n(t_k) - f(t_k)|.$$

Make two runs:

(a) $N = 5, 10, 20$; (b) $N = 4$.

Comment on the results.

4. Write a program for discrete polynomial least squares approximation of a function f defined on [–1,1], using the inner product

$$(u, v) = \frac{2}{N+1}\sum_{i=0}^{N} u(t_i)v(t_i),\ \ t_i = -1 + \frac{2i}{N}.$$

Follow these steps.

(a) The recurrence coefficients for the appropriate (monic) orthogonal polynomials $\{\pi_k(t)\}$ are known explicitly:

$$\alpha_k = 0,\ \ k = 0, 1, \ldots, N;\quad \beta_0 = 2,$$

$$\beta_k = \left(1 + \frac{1}{N}\right)^2 \left(1 - \left(\frac{k}{N+1}\right)^2\right)\left(4 - \frac{1}{k^2}\right)^{-1},\ \ k = 1, 2, \ldots, N.$$

(You do not have to prove this.) Define $\gamma_k = \|\pi_k\|^2 = (\pi_k, \pi_k)$, which is known to be equal to $\beta_0\beta_1 \cdots \beta_k$ (cf. Ex. 20).

(b) Using the recurrence formula with coefficients α_k, β_k given in (a), generate an array $\boldsymbol{\pi}$ of dimension $(N+2, N+1)$ containing $\pi_k(t_\ell)$, $k = 0, 1, \ldots, N+1$; $\ell = 0, 1, \ldots, N$. (Here k is the row index and ℓ the column index.) Define $\mu_k = \max_{0 \le \ell \le N} |\pi_k(t_\ell)|$, $k = 1, 2, \ldots, N+1$. Print β_k, γ_k, and μ_{k+1} for $k = 0, 1, 2, \ldots, N$, where $N = 10$. Comment on the results.

(c) With $\hat{p}_n(t) = \sum_{k=0}^{n} \hat{c}_k \pi_k(t)$, $n = 0, 1, \ldots, N$, denoting the least squares approximation of degree $\le n$ to the function f on [–1,1], define

$$\|e_n\|_2 = \|\hat{p}_n - f\| = (\hat{p}_n - f, \hat{p}_n - f)^{1/2},$$
$$\|e_n\|_\infty = \max_{0 \le i \le N} |\hat{p}_n(t_i) - f(t_i)|.$$

Using the array $\boldsymbol{\pi}$ generated in part (b), compute $\hat{c}_n$, $\|e_n\|_2$, $\|e_n\|_\infty$, $n = 0, 1, \ldots, N$, for the following four functions:

$$f(t) = \mathrm{e}^{-t}, \quad f(t) = \ln(2+t), \quad f(t) = \sqrt{1+t}, \quad f(t) = |t|.$$

Be sure you compute $\|e_n\|_2$ as accurately as possible. For $N = 10$ and for each f, print $\hat{c}_n$, $\|e_n\|_2$, and $\|e_n\|_\infty$ for $n = 0, 1, 2, \ldots, N$. Comment on your results. In particular, from the information provided in the output, discuss to what extent the computed coefficients $\hat{c}_k$ may be corrupted by rounding errors.

5. (a) A Sobolev-type least squares approximation problem results if the inner product is defined by

$$(u, v) = \int_{\mathbb{R}} u(t)v(t)\mathrm{d}\lambda_0(t) + \int_{\mathbb{R}} u'(t)v'(t)\mathrm{d}\lambda_1(t),$$

where $\mathrm{d}\lambda_0$, $\mathrm{d}\lambda_1$ are positive measures. What does this type of approximation try to accomplish?

(b) Letting $\mathrm{d}\lambda_0(t) = \mathrm{d}t$, $\mathrm{d}\lambda_1(t) = \lambda\mathrm{d}t$ on [0,2], where $\lambda > 0$ is a parameter, set up the normal equations for the Sobolev-type approximation in (a) of the function $f(t) = \mathrm{e}^{-t^2}$ on [0,2] by means of a polynomial of degree $n-1$. Use the basis $\pi_j(t) = t^{j-1}$, $j = 1, 2, \ldots, n$. {*Hint*: express the components b_i of the right-hand vector of the normal equations in terms of the "incomplete gamma function" $\gamma(a, x) = \int_0^x t^{a-1}\mathrm{e}^{-t}\mathrm{d}t$ with $x = 4$, $a = i/2$.}

(c) Use Matlab to solve the normal equations for $n = 2:5$ and $\lambda = 0, .5, 1, 2$. Print

$$\|\hat{\varphi}_n - f\|_\infty \quad \text{and} \quad \|\hat{\varphi}_n' - f'\|_\infty, \quad n = 2, 3, 4, 5$$

(or a suitable approximation thereof) along with the condition numbers of the normal equations. {Use the following values for the incomplete gamma function: $\gamma(\frac{1}{2},4) = 1.764162781524843$, $\gamma(1,4) = 0.9816843611112658$, $\gamma(\frac{3}{2},4) = 0.8454501129849537$, $\gamma(2,4) = 0.9084218055563291$, $\gamma(\frac{5}{2},4) = 1.121650058367554$.} Comment on the results.

6. With $\omega_n(x) = \prod_{k=0}^{n}(x-k)$, let M_n be the largest, and m_n the smallest, relative maximum of $|\,\omega_n(x)|$. For $n = 5:5:30$ calculate M_n, m_n, and M_n/m_n, using Newton's method (cf. Chap 4, Sect. 4.6), and print also the respective number of iterations.
7. (a) Write a subroutine that produces the value of the interpolation polynomial $p_n(f; x_0, x_1, \ldots, x_n; t)$ at any real t, where $n \geq 0$ is a given integer, x_i are $n+1$ distinct nodes, and f is any function available in the form of a function subroutine. Use Newton's interpolation formula and exercise frugality in the use of memory space when generating the divided differences. It is possible, indeed, to generate them "in place" in a single array of dimension $n+1$ that originally contains the values $f(x_i)$, $i = 0, 1, \ldots, n$. {*Hint*: generate the divided differences from the bottom up.}
   (b) Run your routine on the function $f(t) = \frac{1}{1+t^2}$, $-5 \leq t \leq 5$, using $x_i = -5 + 10\frac{i}{n}$, $i = 0, 1, \ldots, n$, and $n = 2:2:8$ (Runge's example). Plot the polynomials against the exact function.
8. (a) Write a Matlab function `y=tridiag(n,a,b,c,v)` for solving a tridiagonal (nonsymmetric) system

$$\begin{bmatrix} a_1 & c_1 & & & 0 \\ b_1 & a_2 & c_2 & & \\ & b_2 & a_3 & c_3 & \\ & & \ddots & \ddots & \ddots \\ & & & & c_{n-1} \\ 0 & & & b_{n-1} & a_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_{n-1} \\ y_n \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_{n-1} \\ v_n \end{bmatrix}$$

   by Gauss elimination without pivoting. Keep the program short.
   (b) Write a program for computing the natural spline interpolant $s_{\mathrm{nat}}(f;\cdot)$ on an arbitrary partition $a = x_1 < x_2 < x_3 < \cdots < x_{n-1} < x_n = b$ of $[a,b]$. Print $\{i, \mathrm{errmax}(i); i = 1, 2, \ldots, n-1\}$, where

$$\mathrm{errmax}(i) = \max_{1 \leq j \leq N} \left|s_{\mathrm{nat}}(f; x_{i,j}) - f(x_{i,j})\right|, \quad x_{i,j} = x_i + \frac{j-1}{N-1}\,\Delta x_i.$$

   (You will need the function `tridiag`.) Test the program for cases in which the error is zero (what are these, and why?).
   (c) Write a second program for computing the complete cubic spline interpolant $s_{\mathrm{compl}}(f;\cdot)$ by modifying the program in (b) with a minimum of changes. Highlight the changes in the program listing. Apply (and justify) a test similar to that of (b).

(d) Run the programs in (b) and (c) for $[a, b] = [0, 1]$, $n = 11$, $N = 51$, and

(i) $x_i = \frac{i-1}{n-1}$, $i = 1, 2, \ldots, n$; $f(x) = \mathrm{e}^{-x}$ and $f(x) = x^{5/2}$;

(ii) $x_i = \left(\frac{i-1}{n-1}\right)^2$, $i = 1, 2, \ldots, n$; $f(x) = x^{5/2}$.

Comment on the results.

Selected Solutions to Exercises

11. (a) We have

$$(\pi_r, \pi_s) = \int_0^\infty \mathrm{e}^{-(r+s)t}\,\mathrm{d}t = -\frac{1}{r+s}\,\mathrm{e}^{-(r+s)t}\Big|_0^\infty = \frac{1}{r+s},$$

$$(\pi_r, f) = \int_0^1 \mathrm{e}^{-rt}\,\mathrm{d}t = -\frac{1}{r}\,\mathrm{e}^{-rt}\Big|_0^1 = \frac{1}{r}(1 - \mathrm{e}^{-r}).$$

The normal equations, therefore, are

$$\sum_{s=1}^{n} \frac{1}{r+s}\, c_s = \frac{1}{r}\,(1 - \mathrm{e}^{-r}), \quad r = 1, 2, \ldots, n.$$

The matrix is the Hilbert matrix of order $n + 1$ with the first column and last row removed.

(b) PROGRAM

```
%EXII_11B
%
f0='%8.0f %12.4e\n';
f1='%45.14e\n';
disp('        n       cond        solution')
for n=1:8
  A=hilb(n+1);
  A(:,1)=[];
  A(n+1,:)=[];
  x=(1:n)';
  b=(1-exp(-x))./x;
  c=A\b;
  cd=cond(A);
  fprintf(f0,n,cd)
  fprintf(f1,c)
  for i=1:201
    t=.01*(i-1);
    fa(i,n)=sum(c.*exp(-x*t));
  end
```

```
end
for i=1:11
  tf(i)=.1*(i-1);
  f(i)=1;
end
for i=1:201
  tfa(i)=.01*(i-1);
end
plot(tf,f);
hold on
plot(ones(size(tf)),tf);
plot(tfa,fa(:,1),':');
plot(tfa,fa(:,2),'-.');
plot(tfa,fa(:,3),'--');
plot(tfa,fa(:,4),'-');
axis([0 2 0 1.5]);
hold off
```

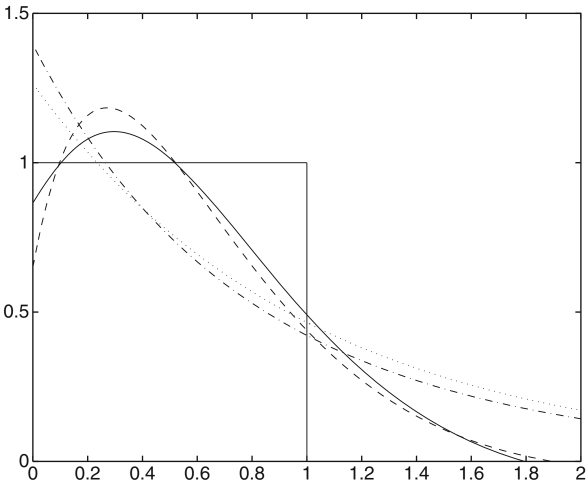
OUTPUT

```
>> EXII_11B
  n      cond              solution
  1   1.0000e+00
                        1.26424111765712e+00
  2   3.8474e+01
                        1.00219345775339e+00
                        3.93071489855589e-01
  3   1.3533e+03
                       -1.23430987802214e+00
                        9.33908483295774e+00
                       -7.45501111925180e+00
  4   4.5880e+04
                       -2.09728726098036e+00
                        1.58114152051443e+01
                       -2.03996718636248e+01
                        7.55105210088422e+00
  5   1.5350e+06
                        2.95960905289307e-01
                       -1.29075627900844e+01
                        8.01167511196597e+01
                       -1.26470845210147e+02
                        6.03098537899591e+01
  6   5.1098e+07
                        2.68879580265092e+00
```

```
                          -5.47821734938751e+01
                           3.03448008206274e+02
                          -6.28966173654415e+02
                           5.62805182233655e+02
                          -1.84248287095832e+02

   7    1.6978e+09
                           1.19410815562677e+00
                          -1.89096699709436e+01
                           3.44042318216034e+01
                           2.67846414188988e+02
                          -9.16935587561045e+02
                           9.99544328640241e+02
                          -3.66412000082403e+02
   8    5.6392e+10
                          -2.39677086853911e+00
                           9.42030165764484e+01
                          -1.09672261269167e+03
                           5.45217770865201e+03
                          -1.33593305457727e+04
                           1.71746576145770e+04
                          -1.11498207678274e+04
                           2.88841304234284e+03

>>
```

The condition numbers here are even a bit larger than the condition numbers of the Hilbert matrices of the same order (cf. Chap. 1, MA 9).

PLOTS



dotted line: n=1, *dashdotted line*: $n = 2$, *dashed line*: n=3,
solid line n=4

16. (a) Let $p(t) = a_0 + a_1 t + a_2 t^2$. Then p satisfies the constraints if and only if $a_0 = 0,\ a_0 + a_1 + a_2 = 1$, that is,

$$p(t) = t^2 + a_1 t(1-t).$$

Therefore, we need to minimize

$$\int_0^1 [f(t) - p(t)]^2 \mathrm{d}t = \int_0^1 [f(t) - t^2 - a_1 t(1-t)]^2 \mathrm{d}t.$$

This is an unconstrained least squares problem for approximating the function $f(t) - t^2$ by a multiple of $\pi_1(t) = t(1-t)$. The normal equation is

$$a_1 \int_0^1 [t(1-t)]^2 \mathrm{d}t = \int_0^1 (f(t) - t^2) t(1-t) \mathrm{d}t,$$

and yields the solution

$$\hat{p}(t) = t^2 + \hat{a}_1 t(1-t),$$

where

$$\hat{a}_1 = \frac{\int_0^1 (f(t) - t^2) t(1-t) \mathrm{d}t}{\int_0^1 [t(1-t)]^2 \mathrm{d}t} = 30 \int_0^1 f(t) t(1-t) \mathrm{d}t - \frac{3}{2}.$$

(b) If $f(t) = t^r$, then

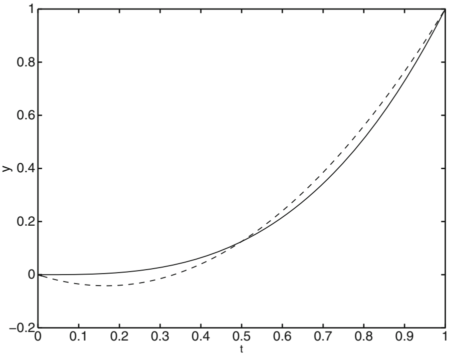
$$\hat{a}_1 = 30 \int_0^1 t^{r+1}(1-t) \mathrm{d}t - \frac{3}{2} = \frac{30}{(r+2)(r+3)} - \frac{3}{2}$$

and

$$\begin{aligned} \hat{p}(t) &= t^2 + \left(\frac{30}{(r+2)(r+3)} - \frac{3}{2} \right) t(1-t) \\ &= t \left\{ \frac{30}{(r+2)(r+3)} - \frac{3}{2} + \left(\frac{5}{2} - \frac{30}{(r+2)(r+3)} \right) t \right\}. \end{aligned}$$

For $r = 3$, this gives $\hat{p}(t) = \frac{1}{2} t(3t - 1)$.

Plot:



solid line: $y = t^3$
dashed line: $y = .5t(3t - 1)$

27. PROGRAM

```
%EXII_27 Lebesgue functions
%
n=5;
%n=10;
%n=20;
i=1:n+1; mu=1:n+1;
% equally spaced points
x=-1+2*(i-1)/n;
%
% Chebyshev points
%x=cos((2*(i-1)+1)*pi/(2*n+2));
%
iplot=0;
for k=2:n+1
%for k=1:n+2
  for j=1:21
    iplot=iplot+1;
    t(iplot)=x(k-1)+(j-1)*(x(k)-x(k-1))/20;
%     if k==1
%       t(iplot)=1+(j-1)*(x(1)-1)/20;
%     elseif k<=n+1
%       t(iplot)=x(k-1)+(j-1)*(x(k)-x(k-1))/20;
%     else
%       t(iplot)=x(n+1)+(j-1)*(-1-x(n+1))/20;
%     end
```

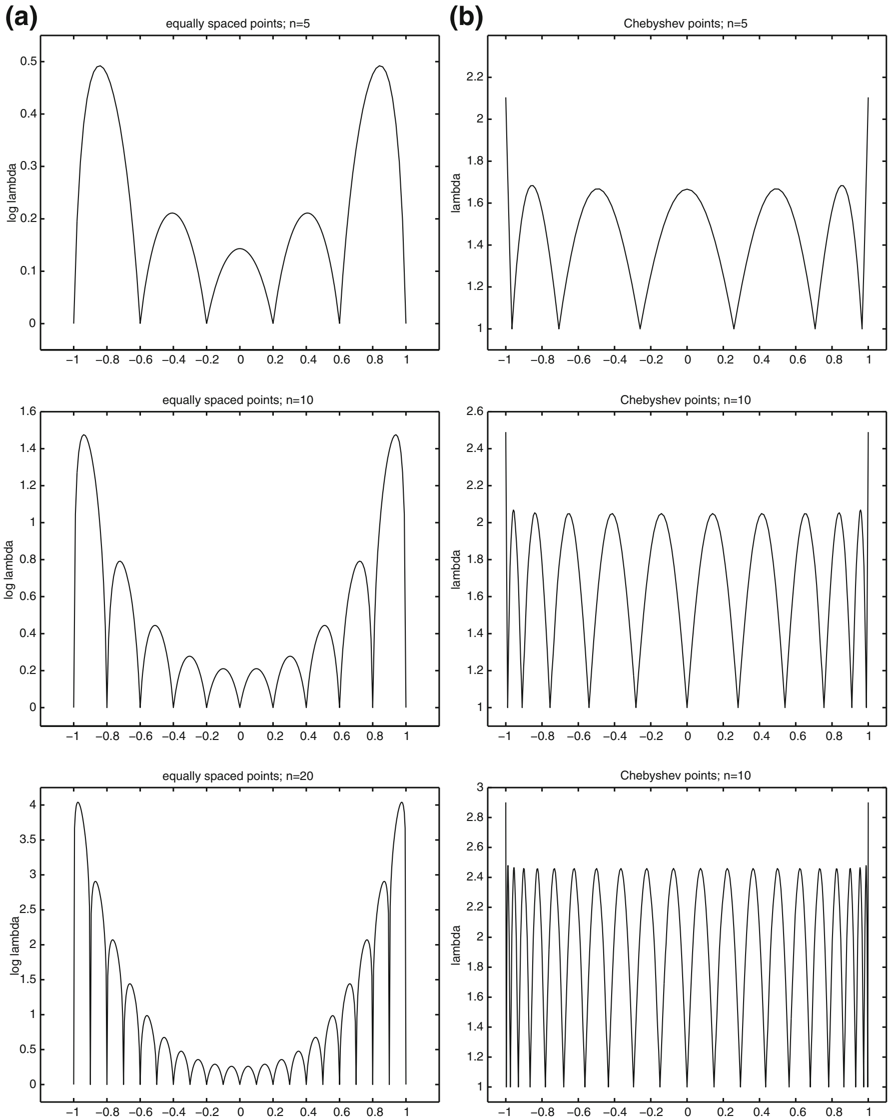
```
      s=0;
      for nu=1:n+1
        mu0=find(mu-nu);
        p=prod((t(iplot)-x(mu0))./(x(nu)-x(mu0)));
        s=s+abs(p);
      end
      leb(iplot)=s;
    end
  end
  plot(t,log10(leb))
  %plot(t,leb)
  axis([-1.2 1.2 -.05 .55])
  %axis([-1.2 1.2 -.1 1.6])
  %axis([-1.2 1.2 -.25 4.25])
  %axis([-1.1 1.1 .9 2.4])
  %axis([-1.1 1.1 .9 2.6])
  %axis([-1.1 1.1 .9 3])
  title('equally spaced points; n=5',
     'Fontsize',14)
  %title('equally spaced points; n=10',
     'Fontsize',14)
  %title('equally spaced points; n=20',
     'Fontsize',14)
  %title('Chebyshev points; n=5','Fontsize',14)
  %title('Chebyshev points; n=10','Fontsize',14)
  ylabel('log lambda','Fontsize',14)
  %ylabel('lambda','Fontsize',14)

   OUTPUT
   (on the next page)
```

At the interpolation nodes x_i, one clearly has $\lambda_n(x_i) = 1$. The local maxima of λ_n between successive interpolation nodes are almost equal, and relatively small, in case (b), but become huge near the endpoints of $[-1, 1]$ in case (a). In case (b), the global maxima occur at the endpoints ± 1.

36. We first prove the assertion of the *Hint*. One easily verifies that the function $\left|\left(x - \frac{i}{n}\right)\left(x - \frac{n-i}{n}\right)\right|$ on $[0, 1]$ is symmetric with respect to the midpoint $\frac{1}{2}$. Being quadratic, its maximum must occur either at $x = 0$ or at $x = \frac{1}{2}$, and hence is the larger of $\frac{i(n-i)}{n^2}$ and $\frac{(n-2i)^2}{4n^2}$. The former attains its maximum at $i = \frac{n}{2}$, the latter at $i = 0$ (and $i = n$). Either one equals $\frac{1}{4}$. Thus,

$$\max_{0 \le x \le 1} \left|\left(x - \frac{i}{n}\right)\left(x - \frac{n-i}{n}\right)\right| \le \frac{1}{4} \quad \text{for } i = 0, 1, \ldots, n,$$

as claimed.



(a) We have

$$e^x - p_n(f;x) = \frac{e^{\xi(x)}}{(n+1)!}\prod_{k=0}^{n}\left(x - \frac{k}{n}\right), \quad 0 < \xi(x) < 1.$$

Here we use

$$\prod_{k=0}^{n}\left|x-\frac{k}{n}\right| = \sqrt{\prod_{i=0}^{n}\left|x-\frac{i}{n}\right|\left|x-\frac{n-i}{n}\right|}$$

along with the assertion of the *Hint* to obtain

$$\max_{0\le x\le 1}|\mathrm{e}^x - p_n(f;x)| \le \frac{\mathrm{e}}{(n+1)!}\left(\frac{1}{4}\right)^{\frac{n+1}{2}} = \frac{\mathrm{e}}{2^{n+1}(n+1)!}.$$

The smallest n making the upper bound $\le 10^{-6}$ is $n = 7$.

(b) From Taylor's formula,

$$\mathrm{e}^x - t_n(x) = \frac{\mathrm{e}^{\xi(x)}}{(n+1)!}\, x^{n+1}, \quad 0 < \xi(x) < 1.$$

Thus,

$$|\mathrm{e}^x - t_n(x)| \le \frac{\mathrm{e}}{(n+1)!}.$$

This bound is larger than the one in (a) by a factor of 2^{n+1}. Accordingly, for it to be $\le 10^{-6}$ now requires $n = 9$.

47. We have, for $0 \le k, \ell < n$,

$$\begin{aligned}(T_k, T_\ell) &= \sum_{\nu=1}^{n} T_k(x_\nu)T_\ell(x_\nu) = \sum_{\nu=1}^{n} \cos\left(k\,\frac{2\nu-1}{2n}\pi\right)\cos\left(\ell\,\frac{2\nu-1}{2n}\pi\right)\\
&= \frac{1}{2}\sum_{\nu=1}^{n}\left[\cos\left((k+\ell)\frac{2\nu-1}{2n}\pi\right) + \cos\left((k-\ell)\frac{2\nu-1}{2n}\pi\right)\right]\\
&= \frac{1}{2}\,\mathrm{Re}\left\{\sum_{\nu=1}^{n}\mathrm{e}^{\mathrm{i}(k+\ell)\frac{2\nu-1}{2n}\pi} + \sum_{\nu=1}^{n}\mathrm{e}^{\mathrm{i}(k-\ell)\frac{2\nu-1}{2n}\pi}\right\}\\
&= \frac{1}{2}\,\mathrm{Re}\left\{\mathrm{e}^{\mathrm{i}(k+\ell)\frac{\pi}{2n}}\sum_{\nu=1}^{n}\mathrm{e}^{\mathrm{i}(k+\ell)\frac{\nu-1}{n}\pi} + \mathrm{e}^{\mathrm{i}(k-\ell)\frac{\pi}{2n}}\sum_{\nu=1}^{n}\mathrm{e}^{\mathrm{i}(k-\ell)\frac{\nu-1}{n}\pi}\right\}.\end{aligned}$$

Assume $k \ne \ell$. Both sums in the last equation are finite geometric series and can thus be summed explicitly. One gets

$$\begin{aligned}(T_k, T_\ell) &= \frac{1}{2}\,\mathrm{Re}\left\{\mathrm{e}^{\mathrm{i}(k+\ell)\frac{\pi}{2n}}\,\frac{1-\mathrm{e}^{\mathrm{i}(k+\ell)\pi}}{1-\mathrm{e}^{i\frac{k+\ell}{n}\pi}} + \mathrm{e}^{\mathrm{i}(k-\ell)\frac{\pi}{2n}}\,\frac{1-\mathrm{e}^{\mathrm{i}(k-\ell)\pi}}{1-\mathrm{e}^{\mathrm{i}\frac{k-\ell}{n}\pi}}\right\}\\
&= \frac{1}{2}\,\mathrm{Re}\left\{\frac{i\,[1-\mathrm{e}^{\mathrm{i}(k+\ell)\pi}]}{2\sin\frac{k+\ell}{2n}\pi} + \frac{i\,[1-\mathrm{e}^{\mathrm{i}(k-\ell)\pi}]}{2\sin\frac{k-\ell}{2n}\pi}\right\},\end{aligned}$$

where the denominators are not zero by the assumption on k and ℓ. Now if $k+\ell$ (and hence also $k-\ell$) is even, both expressions in brackets are zero. If $k+\ell$ (and hence also $k-\ell$) is odd, the numerators are both $2i$, hence the real part equals zero. In either case, $(T_k, T_\ell) = 0$, as claimed.

An easy argument also shows that the value of the inner product is $\frac{n}{2}$ if $k = \ell > 0$, and n if $k = \ell = 0$.

The result follows more easily from the continuous orthogonality (cf. Sect. 2.2.4, (2.99)) by applying the Gauss–Chebyshev quadrature formula (cf. Chap. 3, Sect. 3.2.3 and Ex. 36).

57. (a) For $n = 1$, the assertion of the *Hint* is true for all $r \geq 0$ since

$$[x_r, x_{r+1}] = \frac{\log_{10} x_{r+1} - \log_{10} x_r}{x_{r+1} - x_r} = \frac{(r+1) - r}{10^r(10-1)} = \frac{1}{9 \cdot 10^r}.$$

Thus, assume the assertion to be true for some n and all $r \geq 0$. Then, by the property (2.113) of divided differences,

$$\begin{aligned}
&[x_r, x_{r+1}, \ldots, x_{r+n}, x_{r+n+1}]f \\
&\quad = \frac{[x_{r+1}, x_{r+2}, \ldots, x_{r+n+1}]f - [x_r, x_{r+1}, \ldots, x_{r+n}]f}{x_{r+n+1} - x_r} \\
&\quad = \frac{(-1)^{n-1}}{10^{rn+n(n-1)/2}(10^n - 1)} \, \frac{1 - 10^n}{10^n(10^{r+n+1} - 10^r)} \\
&\quad = \frac{(-1)^n}{10^{rn+n(n-1)/2}\, 10^{n+r}(10^{n+1} - 1)} \\
&\quad = \frac{(-1)^n}{10^{r(n+1)+n(n+1)/2}\,(10^{n+1} - 1)},
\end{aligned}$$

which is precisely the assumed assertion with n replaced by $n+1$.

(b) Let $a_k = [x_0, x_1, \ldots, x_k]f$. By Newton's formula, noting that $a_0 = \log_{10} 1 = 0$, we have

$$\begin{aligned}
p_n(x) &= \sum_{k=1}^{n} a_k (x-1)(x-10)\cdots(x-10^{k-1}) \\
&= \sum_{k=1}^{n} \frac{(-1)^{k-1}}{10^{k(k-1)/2}(10^k - 1)} \prod_{\ell=0}^{k-1}(x - 10^\ell) \\
&= -\sum_{k=1}^{n} \frac{1}{10^{k(k-1)/2}(10^k - 1)} \prod_{\ell=0}^{k-1}(10^\ell - x)
\end{aligned}$$

$$= -\sum_{k=1}^{n} \frac{1}{10^k - 1} \prod_{\ell=0}^{k-1} (1 - x/10^\ell)$$

$$= -\sum_{k=1}^{n} t_k(x),$$

where

$$t_k(x) := \frac{1}{10^k - 1} \prod_{\ell=0}^{k-1} (1 - x/10^\ell).$$

For $1 \le x < 10$, we have

$$|t_k(x)| < \left| \frac{(1-x)(1-x/10^{k-1})}{10^k - 1} \right| < \frac{9}{10^k - 1}\left(1 - \frac{1}{10^{k-1}}\right) < \frac{9}{10^k}.$$

Thus, the infinite series $\sum_{k=1}^{\infty} t_k(x)$ is majorized by the convergent geometric series $9\sum_{k=1}^{\infty} 10^{-k}$ and therefore also converges. However, for $x = 9$, one computes

$$-\sum_{k=1}^{\infty} t_k(9) = \sum_{k=1}^{\infty} \frac{8}{10^k - 1} \prod_{\ell=1}^{k-1} (1 - 9/10^k)$$
$$= 0.89777\ldots < \log_{10}(9) = 0.95424\ldots.$$

(For an analysis of the discrepancy, see Gautschi [2008].)

75. Let $s(x) = \sum_{j=1}^{n} c_j B_j(x)$. Then, with points ξ_ν as defined, the first $n-1$ conditions imposed on s can be written as

$$c_1 B_1(\xi_1) + c_2 B_2(\xi_1) = f_1,$$
$$c_2 B_2(\xi_2) + c_3 B_3(\xi_2) = f_2,$$
$$\cdots\cdots\cdots\cdots\cdots\cdots$$
$$c_{n-1} B_{n-1}(\xi_{n-1}) + c_n B_n(\xi_{n-1}) = f_{n-1}.$$

The last condition imposed is, since $B_1(x_1) = B_n(x_n) = 1$,

$$c_1 - c_n = 0.$$

The matrix of the system has the following structure:

$$\begin{bmatrix} \times & \times & & & & \\ & \times & \times & & & \\ & & \times & \times & & \\ & & & \ddots & \ddots & \\ & & & & \times & \times \\ 1 & & & & & -1 \end{bmatrix}$$

Note that $B_j(\xi_j) \neq 0$ for $j = 1, 2, \ldots, n-1$.

Solution: (1) Subtract a suitable multiple of the first equation from the last equation to create a zero in position $(n, 1)$. This produces a fill-in in position $(n, 2)$. (2) Subtract a suitable multiple of the second equation from the last to create a zero in position $(n, 2)$. This produces a fill-in in position $(n, 3)$, etc. After $n-1$ such operations one obtains a nonsingular upper bidiagonal system, which is quickly solved by back substitution.

79. (a) From Sect. 2.2.4, (2.140) and (2.141), the spline on $[x_i, x_{i+1}]$ is

$$(*) \qquad p_i(x) = c_{i,0} + c_{i,1}(x - x_i) + c_{i,2}(x - x_i)^2 + c_{i,3}(x - x_i)^3,$$

where

$$\binom{*}{*} \qquad \begin{aligned} c_{i,0} = f_i, \;\; c_{i,1} = m_i, \;\; c_{i,2} &= \frac{[x_i, x_{i+1}]f - m_i}{\Delta x_i} - c_{i,3}\Delta x_i, \\ c_{i,3} &= \frac{m_{i+1} + m_i - 2[x_i, x_{i+1}]f}{(\Delta x_i)^2}. \end{aligned}$$

The two "not-a-knot" conditions are $p_1'''(x_2) = p_2'''(x_2)$, $p_{n-2}'''(x_{n-1}) = p_{n-1}'''(x_{n-1})$. By (*), this yields

$$c_{1,3} = c_{2,3}, \quad c_{n-2,3} = c_{n-1,3}.$$

Substituting from $\binom{*}{*}$, the first equality becomes

$$\frac{m_2 + m_1 - 2[x_1, x_2]f}{(\Delta x_1)^2} = \frac{m_3 + m_2 - 2[x_2, x_3]f}{(\Delta x_2)^2},$$

or, after some elementary manipulations,

$$(1) \qquad \begin{aligned} & m_1 + \left(1 - \left(\frac{\Delta x_1}{\Delta x_2}\right)^2\right) m_2 - \left(\frac{\Delta x_1}{\Delta x_2}\right)^2 m_3 \\ &= 2\left([x_1, x_2]f - \left(\frac{\Delta x_1}{\Delta x_2}\right)^2 [x_2, x_3]f\right) =: b_1. \end{aligned}$$

Similarly, the second equality becomes

$$(n) \qquad \begin{aligned} & m_{n-2} + \left(1 - \left(\frac{\Delta x_{n-2}}{\Delta x_{n-1}}\right)^2\right) m_{n-1} - \left(\frac{\Delta x_{n-2}}{\Delta x_{n-1}}\right)^2 m_n \\ &= 2\left([x_{n-2}, x_{n-1}]f - \left(\frac{\Delta x_{n-2}}{\Delta x_{n-1}}\right)^2 [x_{n-1}, x_n]f\right) =: b_n. \end{aligned}$$

(b) The first equation (for $i = 2$) from Sect. 2.2.4, (2.145), is

$$(2) \qquad \Delta x_2\, m_1 + 2(\Delta x_1 + \Delta x_2)\, m_2 + \Delta x_1\, m_3 = b_2.$$

Multiply (2) by $\dfrac{\Delta x_1}{(\Delta x_2)^2}$ and add to (1) to get the new pair of equations

$$\begin{cases} \left(1 + \dfrac{\Delta x_1}{\Delta x_2}\right) m_1 + \left(1 + \dfrac{\Delta x_1}{\Delta x_2}\right)^2 m_2 = b_1 + \dfrac{\Delta x_1}{(\Delta x_2)^2}\, b_2, \\ \Delta x_2\, m_1 + 2(\Delta x_1 + \Delta x_2)\, m_2 + \Delta x_1\, m_3 = b_2. \end{cases}$$

This is the beginning of a tridiagonal system.

Similarly, the last equation (for $i = n-1$) from Sect. 2.2.4, (2.145), is

$$(n-1) \quad \Delta x_{n-1}\, m_{n-2} + 2(\Delta x_{n-2} + \Delta x_{n-1})\, m_{n-1} + \Delta x_{n-2}\, m_n = b_{n-1}.$$

Multiply Eq. $(n-1)$ by $\frac{1}{\Delta x_{n-1}}$ and subtract from (n); then the last two equations become

$$\begin{cases} \Delta x_{n-1}\, m_{n-2} + 2(\Delta x_{n-2} + \Delta x_{n-1})\, m_{n-1} + \Delta x_{n-2}\, m_n = b_{n-1} \\ -\left(1 + \dfrac{\Delta x_{n-2}}{\Delta x_{n-1}}\right)^2 m_{n-1} - \dfrac{\Delta x_{n-2}}{\Delta x_{n-1}}\left(1 + \dfrac{\Delta x_{n-2}}{\Delta x_{n-1}}\right) m_n = b_n - \dfrac{1}{\Delta x_{n-1}}\, b_{n-1}. \end{cases}$$

This is the end of the tridiagonal system.

(c) No: the system is *not* diagonally dominant, since in the first equation the diagonal element $1 + \frac{\Delta x_1}{\Delta x_2}$ is less than the other remaining element $\left(1 + \frac{\Delta x_1}{\Delta x_2}\right)^2$.

Selected Solutions to Machine Assignments

4. (a), (b) and (c)

```
 PROGRAMS

%MAII_4ABC
%
%(a)
%
function [beta,gamma,mu,coeff,L2err, ...
   maxerr]=MAII_4ABC(N)
P=zeros(N+2,N+1); i=0:N; t=-1+2*i/N;
beta(1)=2; b=(1+1/N)^2; gamma(1)=2;
for k=1:N
  beta(k+1)=b*(1-(k/(N+1))^2) ...
    /(4-1/k^2);
  gamma(k+1)=beta(k+1)*gamma(k);
end
%
%(b) and (c)
%
P(1,:)=1; P(2,:)=t; mu(1)=max(abs(t));
for k=2:N+1
  P(k+1,:)=t.*P(k,:)-beta(k)*P(k-1,:);
  mu(k)=max(abs(P(k+1,:)));
end
for n=0:N
  coeff(n+1)=2*sum(P(n+1,:) ...
     .*f(t))/((N+1)*gamma(n+1));
end
for n=0:N
  emax=0; e2=0;
  for k=1:N+1
    e=abs(sum(coeff(1:n+1)' ...
      .*P(1:n+1,k))-f(t(k)));
    if e>emax, emax=e; end
    e2=e2+e^2;
  end
  L2err(n+1)=sqrt(2*e2/(N+1));
  maxerr(n+1)=emax;
end

function y=f(x)
y=exp(-x);
```

```
%y=log(2+x);
%y=sqrt(1+x);
%y=abs(x);

%RUNMAII_4ABC  Driver program for
% MAII_4ABC
%
f0='%4.0f %20.15f %23.15e %23.15e\n';
f1='%12.0f %23.15e %12.4e %12.4e\n';
disp(['   k          beta(k)' ...
  '                  gamma(k)' ...
  '                   mu(k+1)'])
N=10;
[beta,gamma,mu,coeff,L2err,maxerr] ...
  =MAII_4ABC(N);
for k=1:N+1
  fprintf(f0,k-1,beta(k),gamma(k), ...
  mu(k))
end
fprintf('\n')
disp(['        n      coefficients' ...
  '      L2 error     max error'])
for k=1:N+1
  fprintf(f1,k-1,coeff(k),L2err(k), ...
  maxerr(k))
end
```

OUTPUT

```
>> runMAII_4ABC
   k         beta(k)               gamma(k)                  mu(k+1)
   0    2.000000000000000   2.000000000000000e+00   1.000000000000000e+00
   1    0.400000000000000   8.000000000000002e-01   5.999999999999999e-01
   2    0.312000000000000   2.496000000000001e-01   2.879999999999998e-01
   3    0.288000000000000   7.188480000000004e-02   1.152000000000001e-01
   4    0.266666666666667   1.916928000000001e-02   7.680000000000001e-02
   5    0.242424242424242   4.647098181818186e-03   3.351272727272728e-02
   6    0.213986013986014   9.944140165289268e-04   1.488738461538463e-02
   7    0.180923076923077   1.799124436058489e-04   5.269231888111895e-03
   8    0.143058823529412   2.573806252055439e-05   1.834253163307282e-03
   9    0.100309597523220   2.581774692464279e-06   5.068331109138542e-04
  10    0.052631578947368   1.358828785507516e-07   3.771414197649772e-17

   n       coefficients         L2 error       max error
   0   1.212203623058161e+00   1.0422e+00      1.5061e+00    f(t)=exp(-t)
   1  -1.123748299778268e+00   2.7551e-01      3.8233e-01
   2   5.430255798492442e-01   4.7978e-02      5.6515e-02
   3  -1.774967744700318e-01   6.0942e-03      5.7637e-03
   4   4.380735440218084e-02   5.9354e-04      7.1705e-04
   5  -8.681309127655327e-03   4.5392e-05      5.0328e-05
```

```
  6   1.436822757642024e-03   2.7410e-06     3.1757e-06
  7  -2.041262174764023e-04   1.2894e-07     1.3676e-07
  8   2.539983220653184e-05   4.5185e-09     5.2244e-09
  9  -2.811356175489033e-06   1.0329e-10     1.4201e-10
 10   2.800374538854939e-07   6.2576e-14     7.9492e-14

  n        coefficients         L2 error     max error
  0   6.379455015198038e-01   4.8331e-01   6.3795e-01      f(t)=ln(2+t)
  1   5.341350646266596e-01   7.3165e-02   1.0381e-01
  2  -1.436962628313260e-01   1.4112e-02   1.7593e-02
  3   5.149163971020859e-02   2.9261e-03   3.2053e-03
  4  -2.066713722106771e-02   6.1199e-04   8.2444e-04
  5   8.790920250468457e-03   1.2409e-04   1.4929e-04
  6  -3.863610725685766e-03   2.3550e-05   3.0261e-05
  7   1.730112538790927e-03   4.0112e-06   4.5316e-06
  8  -7.825109066954439e-04   5.7410e-07   6.9079e-07
  9   3.553730415235034e-04   5.9485e-08   8.1789e-08
 10  -1.613718817644612e-04   2.1657e-14   2.7534e-14

  n        coefficients         L2 error     max error
  0   9.134654065768736e-01   5.7547e-01   9.1347e-01    f(t)=sqrt(1+t)
  1   6.165636213969754e-01   1.6444e-01   2.9690e-01
  2  -2.799173478370132e-01   8.6512e-02   1.2895e-01
  3   2.654751232178156e-01   4.9173e-02   7.8888e-02
  4  -2.969055755002559e-01   2.6985e-02   4.4684e-02
  5   3.416320385824934e-01   1.3632e-02   1.8447e-02
  6  -3.862066935480817e-01   6.1238e-03   9.0935e-03
  7   4.215059528049150e-01   2.3530e-03   3.0774e-03
  8  -4.411293520434504e-01   7.2661e-04   9.1223e-04
  9   4.416771232533437e-01   1.5591e-04   2.1437e-04
 10  -4.229517654415031e-01   2.7984e-14   3.5305e-14

  n        coefficients         L2 error     max error
  0   5.454545454545454e-01   4.5272e-01   5.4545e-01    f(t)=|t|
  1   5.046468293750710e-17   4.5272e-01   5.4545e-01
  2   8.741258741258736e-01   1.1933e-01   1.9580e-01
  3   0.000000000000000e+00   1.1933e-01   1.9580e-01
  4  -7.284382284382317e-01   6.3786e-02   1.1189e-01
  5  -5.429698379835253e-16   6.3786e-02   1.1189e-01
  6   1.531862745098003e+00   4.1655e-02   6.9107e-02
  7  -3.506197370654419e-15   4.1655e-02   6.9107e-02
  8  -6.118812656642364e+00   2.7776e-02   3.8191e-02
  9   2.061545898679865e-13   2.7776e-02   3.8191e-02
 10   7.535204475298005e+01   3.9494e-14   5.0709e-14
>>
```

Comments

- Note that the last entry in the `mu` column vanishes, confirming that π_{N+1} vanishes at all the $N+1$ nodes t_ℓ (cf. Ex. 22(b)).
- The calculation of $\hat{c}_n$ is subject to severe cancellation errors as n increases. Indeed, from the formula for the coefficient $\hat{c}_n$ (cf. (2.24)),

$$\hat{c}_n = 2\sum_{\ell=0}^{N} f(t_\ell)\pi_n(t_\ell)/((N+1)\gamma_n)),$$

one expects $\gamma_n \cdot \hat{c}_n$, being a "mean value" of the quantities $\pi_n(t_\ell)$, to have the order of magnitude of these quantitites, i.e., of μ_n, unless there is considerable cancellation in the summation, in which case $\gamma_n \cdot \hat{c}_n$ is much smaller in absolute value than μ_n. That, in fact, is clearly observed in our output when n gets large.

- The maximum error for $n = N$ should be zero since π_N interpolates. This is confirmed reasonably well in the output.
- e^{-t}: Note the rapid convergence. This is because the exponential function is an entire function, hence very smooth.
- $\ln(2 + t)$: Remarkably good convergence in spite of the logarithmic singularity at $t = -2$, a distance of 1 from the left endpoint of [–1,1].
- $\sqrt{1+t}$: Slow convergence because of $f'(t) \to \infty$ as $t \to -1$. There is a branch-point singularity at $t = -1$.
- $|t|$: Extremely slow convergence since f is not differentiable at $t = 0$. Since f is *even*, the approximation for n odd is exactly the same as the one for the preceding even n. This is evident from the L_2 and maximum errors and from the vanishing of the odd-numbered coefficients.

7. (a) PROGRAM

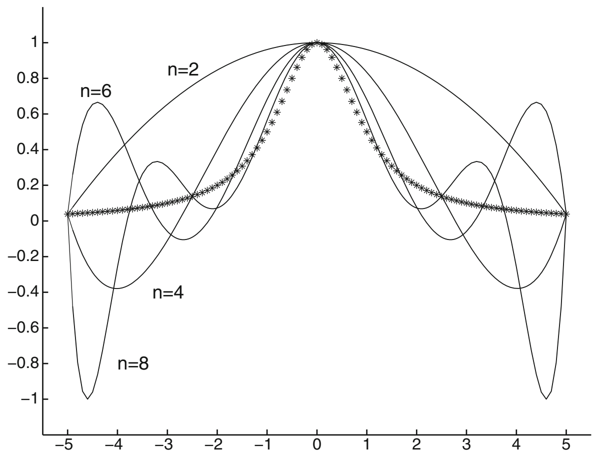
```
%MAII_7AB
%
hold on
it=(0:100)'; t=-5+it/10;
y=1./(1+t.^2);
plot(t,y,'k*')
axis([-5.5 5.5 -1.2 1.2])
for n=2:2:8;
  i=(0:n)'; it=(0:100)';
  x=-5+10*i/n; t=-5+it/10;
  y=pnewt(n,x,t);
  plot(t,y)
end
hold off

%PNEWT
%
function y=pnewt(n,x,t)
d=zeros(n+1,1);
d=f(x);
if n==0
  y=d(1);
  return
end
for j=1:n
```

```
  for i=n:-1:j
    d(i+1)=(d(i+1)-d(i))/(x(i+1)-x(i+1-j));
  end
end
y=d(n+1);
for i=n:-1:1
  y=d(i)+(t-x(i)).*y;
end

function y=f(x)
y=1./(1+x.^2);
```

(b)

```
 OUTPUT
```

The interpolation polynomials are drawn as solid lines, the exact function as black stars.



The "Runge phenomenon", i.e., the violent oscillations of the interpolants near the end points, is clearly evident.

8. (a)

```
PROGRAM

%TRIDIAG
%
%   Gauss elimination without pivoting for a nxn (not
%   necessarily symmetric) tridiagonal system with nonzero
%   diagonal elements a, subdiagonal elements b, superdiagonal
%   element c, and right-hand vector v. The solution vector
%   is y. The vectors a and v will undergo changes by the
%   routine.
%
```

```
function y=tridiag(n,a,b,c,v)
y=zeros(n,1);
for i=2:n
  r=b(i-1)/a(i-1);
  a(i)=a(i)-r*c(i-1);
  v(i)=v(i)-r*v(i-1);
end
y(n)=v(n)/a(n);
for i=n-1:-1:1
  y(i)=(v(i)-c(i)*y(i+1))/a(i);
end
```

(b) The natural spline on the interval $[x_i, x_{i+1}]$ is (cf. (2.140), (2.141))

$$s_{\text{nat}}(x) = c_{i,0} + c_{i,1}(x - x_i) + c_{i,2}(x - x_i)^2 + c_{i,3}(x - x_i)^3, \quad x_i \le x \le x_{i+1},$$

where

$$\begin{aligned} c_{i,0} &= f_i, \\ c_{i,1} &= m_i, \\ c_{i,2} &= \frac{[x_i, x_{i+1}]f - m_i}{\Delta x_i} - c_{i,3}\Delta x_i, \\ c_{i,3} &= \frac{m_{i+1} + m_i - 2[x_i, x_{i+1}]f}{(\Delta x_i)^2}, \end{aligned}$$

and the vector $\boldsymbol{m} = [m_1, m_2, \dots, m_n]^{\mathrm{T}}$ satisfies the tridiagonal system of equations (cf. Sect. 2.2.4, (b.3))

$$\begin{aligned} 2m_1 + m_2 &= b_1 \\ (\Delta x_2)m_1 + 2(\Delta x_1 + \Delta x_2)m_2 + (\Delta x_1)m_3 &= b_2 \\ \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad &\cdots\cdots \\ (\Delta x_{n-1})m_{n-2} + 2(\Delta x_{n-2} + \Delta x_{n-1})m_{n-1} + (\Delta x_{n-2})m_n &= b_{n-1} \\ m_{n-1} + 2m_n &= b_n \end{aligned}$$

where

$$\begin{aligned} b_1 &= 3[x_1, x_2]f \\ b_2 &= 3\{[(\Delta x_2)[x_1, x_2]f + (\Delta x_1)[x_2, x_3]f\} \\ \cdots \quad \cdots \quad &\cdots \quad \cdots \\ b_{n-1} &= 3\{(\Delta x_{n-1})[x_{n-2}, x_{n-1}]f + (\Delta x_{n-2})[x_{n+1}, x_n]f\} \\ b_n &= 3[x_{n-1}, x_n]f \end{aligned}$$

```
 PROGRAM (for natural spline)

%MAII_8B
%
f0='%8.0f %12.4e\n';
n=11; N=51;
a=zeros(n,1); b=zeros(n-1,1); c=b;
i=(1:n)'; j=(1:N)';
x=(i-1)/(n-1);
%x=((i-1)/(n-1)).^2;
f=exp(-x);
%f=sqrt(x).^5;
dx=x(2:n)-x(1:n-1); df=(f(2:n)-f(1:n-1))./dx;
a(1)=2; a(n)=2; b(n-1)=1; c(1)=1;
v(1)=3*df(1); v(n)=3*df(n-1);
a(2:n-1)=2*(dx(1:n-2)+dx(2:n-1));
b(1:n-2)=dx(2:n-1); c(2:n-1)=dx(1:n-2);
v(2:n-1)=3*(dx(2:n-1).*df(1:n-2)+dx(1:n-2).*df(2:n-1));
m=tridiag(n,a,b,c,v);
c0=f(1:n-1); c1=m(1:n-1);
c3=(m(2:n)+m(1:n-1)-2*df)./(dx.^2);
c2=(df-m(1:n-1))./dx-c3.*dx;
emax=zeros(n-1,1);
for i=1:n-1
  xx=x(i)+((j-1)/(N-1))*dx(i);
  t=xx-x(i);
  s=c3(i);
  s=t.*s+c2(i);
  s=t.*s+c1(i);
  s=t.*s+c0(i);
  emax(i)=max(abs(s-exp(-xx)));
%  emax(i)=max(abs(s-sqrt(xx).^5));
  fprintf(f0,i,emax(i))
end
```

(c) For the complete spline, only two small changes need to be made, as indicated by comment lines in the program below.

```
 PROGRAM (for complete spline)

%MAII_8C
%
f0='%8.0f %12.4e\n';
n=11; N=51;
a=zeros(n,1); b=zeros(n-1,1); c=b;
i=(1:n)'; j=(1:N)';
x=(i-1)/(n-1);
% x=((i-1)/(n-1)).^2;
f=exp(-x);
```

```
   %f=sqrt(x).^5;
%
% The next statement is new and does not occur in the
% program of (b)
%
   fder_1=-1; fder_n=-exp(-1);
   %fder_1=0; fder_n=5/2;
   dx=x(2:n)-x(1:n-1); df=(f(2:n)-f(1:n-1))./dx;
%
% The next two lines differ from the corresponding lines
% in the program of (b)
%
   a(1)=1; a(n)=1; b(n-1)=0; c(1)=0;
   v(1)=fder_1; v(n)=fder_n;
   a(2:n-1)=2*(dx(1:n-2)+dx(2:n-1));
   b(1:n-2)=dx(2:n-1); c(2:n-1)=dx(1:n-2);
   v(2:n-1)=3*(dx(2:n-1).*df(1:n-2)+dx(1:n-2).*df(2:n-1));
   m=tridiag(n,a,b,c,v);
   c0=f(1:n-1); c1=m(1:n-1);
   c3=(m(2:n)+m(1:n-1)-2*df)./(dx.^2);
   c2=(df-m(1:n-1))./dx-c3.*dx;
   emax=zeros(n-1,1);
   for i=1:n-1
     xx=x(i)+((j-1)/(N-1))*dx(i);
     t=xx-x(i);
     s=c3(i);
     s=t.*s+c2(i);
     s=t.*s+c1(i);
     s=t.*s+c0(i);
     emax(i)=max(abs(s-exp(-xx)));
   %  emax(i)=max(abs(s-sqrt(xx).^5));
     fprintf(f0,i,emax(i))
   end
```

(d) OUTPUT

```
>> MAII_8B                               >> MAII_8C
  1   4.9030e-04  f(x)=exp(-x)            2.5589e-07  f(x)=exp(-x)
  2   1.3163e-04  natural                 2.2123e-07  complete
  3   3.5026e-05  spline                  2.0294e-07  spline
  4   9.5467e-06  uniform                 1.8288e-07  uniform
  5   2.2047e-06  partition               1.6568e-07  partition
  6   4.2094e-07                          1.4984e-07
  7   3.4559e-06                          1.3568e-07
  8   1.2809e-05                          1.2247e-07
  9   4.8441e-05                          1.1190e-07
 10   1.8036e-04                          9.7227e-08
>>                                       >>
>> MAII_8B                               >> MAII_8C
```

```
  1   2.0524e-04 f(x)=x^(5/2)          4.4346e-05 f(x)=x^(5/2)
  2   5.3392e-05 natural               9.9999e-06 complete
  3   1.6192e-05 spline                4.7121e-06 spline
  4   2.7607e-06 uniform               3.9073e-07 uniform
  5   1.2880e-06 partition             9.8538e-07 partition
  6   9.8059e-06                       5.2629e-07
  7   3.4951e-05                       4.7131e-07
  8   1.3252e-04                       3.6500e-07
  9   4.9310e-04                       3.1193e-07
 10   1.8416e-03                       2.4850e-07
>>                                     >>
>> MAII_8B                             >> MAII_8C
  1   6.6901e-07 f(x)=x^(5/2)          1.0809e-07 f(x)=x^(5/2)
  2   2.3550e-07 natural               6.0552e-07 complete
  3   1.1749e-06 spline                9.1261e-07 spline
  4   1.6950e-06 nonuniform            1.3558e-06 nonuniform
  5   1.5853e-06 partition             1.7319e-06 partition
  6   1.6441e-05                       2.1329e-06
  7   6.8027e-05                       2.5242e-06
  8   3.2950e-04                       2.9138e-06
  9   1.4755e-03                       3.3225e-06
 10   6.5448e-03                       3.6393e-06
>>                                     >>
```

Comments

- Testing: For the natural spline, the error should be exactly zero if f is any linear function. (Not for arbitrary cubics, since f'' does not vanish at $x = 0$ and $x = 1$, unless f is linear.) For the complete spline, the error is zero for any cubic, if one sets $m_1 = f'(0)$ and $m_n = f'(1)$. Example: $f(x) = x^3,\ m_1 = 0,\ m_n = 3$.
- The natural spline for the uniform partition is relatively inaccurate near the endpoints, as expected.
- The complete spline is uniformly accurate for $f(x) = \mathrm{e}^{-x}$ but still relatively inaccurate near $x = 0$ for $f(x) = x^{5/2}$ on account of the "square root" singularity (of f''') at $x = 0$.
- Nonuniform partition (for $f(x) = x^{5/2}$): The natural spline is accurate near $x = 0$ because of the nodes being more dense there, but is still inaccurate at the other end. The complete spline is remarkably accurate at both ends, as well as elsewhere.

http://www.springer.com/978-0-8176-8258-3

Numerical Analysis
Gautschi, W.
2012, XXVI, 588 p. 59 illus., Hardcover
ISBN: 978-0-8176-8258-3
A product of Birkhäuser Basel