Chapter 9
Approximation of Integrals

The usual MATLAB commands for the computation of integrals are quad and quadl. Let us test quad on $\int_0^1 f_0(x)dx = 5/18$, where $f_0(x) := |x - 1/3|$. We obtain

```
>> quad('f0',0,1)-5/18
ans =
   -3.2746e-07
```

With quadl, we find

```
>> quadl('f0',0,1)-5/18
ans =
   -8.9150e-08
```

These errors are far from the eps, the floating-point relative accuracy in MATLAB. With the call quad(fun,a,b,tol), one can modify the tolerance and use an absolute error tolerance tol instead of the default which is 1.0e-6. With tol $= 10^{-15}$ we obtain

```
>> quad('f0',0,1,1e-15)-5/18
ans =
     -3.2196e-15
```

In this chapter, we first test different elementary quadrature formulas of the form $\sum_{i=1}^{n} w_i f(x_i)$ for the approximation of $\int_a^b f(t)\, dt$. We say that a quadrature rule has (exact) **degree of precision** d if it is exact for all polynomials of degree $\leq d$ and not exact for some polynomial of degree $d+1$.

In Exercise 9.1, instead of computing the integral of the function, we compute the integral of its Lagrange interpolant p_n of degree $n-1$ (See Chap. 6). This is tested for small values of n with the use of MATLAB to find the degree of precision in Exercise 9.2. In Exercise 9.4, we show how to improve the degree of precision by a "good" choice of the sampled points, while Exercise 9.5 is a detour to compute a double integral.

T. Lyche and J.-L. Merrien, *Exercises in Computational Mathematics with MATLAB*, Problem Books in Mathematics, DOI 10.1007/978-3-662-43511-3_9,

Then, we are concerned with the approximation of $\int_a^b w(t)f(t)\,dt$ where $w(t)$ is a fixed **weight function**. After a brief review, inner products are used in Exercises 9.6 and 9.7 to obtain accurate results. In Exercise 9.8, we compare two approximations.

A section is devoted to the Monte-Carlo method that uses random numbers to approximate multiple integrals. In Exercise 9.9, we compute an approximation of π by counting random points in a disk and in Exercise 9.10, we compute volumes of ellipsoids, and also intersections and unions of ellipsoids.

We go back to the trapezoidal rule which was introduced at the beginning of the chapter. We first study the basic rule in Exercise 9.11, and then the composite rule in Exercise 9.12. A MATLAB program is given in Exercise 9.13. Finally an approximation to the solution of an integral equation is computed in Exercise 9.14.

The Sect. 9.4 begins with Exercise 9.15 giving the proof of the Euler-Maclaurin formula. This formula provides the bases for the Romberg method (Exercise 9.16) which gives improved approximations starting with the composite trapezoidal rule.

We present a global quadrature rule in Exercise 9.17 that is used to solve an integral equation in Exercise 9.18.

♣**Review:** The following notation and terminology are used:

1. $\mathbb{P}_n$ is the space of polynomials of degree at most n,
2. **nodes** or **sites**: $\boldsymbol{x} = [x_1, \ldots, x_n]$ where $x_1 < x_2 < \ldots < x_n$
3. **node polynomial** $\Pi(x) := \prod_{i=1}^n (x - x_i)$
4. $I(f) := \int_a^b f(x)\,dx$ or $I(f) := \int_a^b w(x)f(x)\,dx$. These are the integrals to be approximated. The fixed **weight function** w is integrable and nonnegative on $[a, b]$.
5. $J(f) := \sum_{i=1}^n w_i f(x_i)$, n point **quadrature rule** . The w_i are called **weights.**
6. A quadrature rule has **degree of precision** d if $I(p) = J(p)$ for any $p \in \mathbb{P}_d$. The rule has **exact degree of precision** d if it has degree of precision d and $I(p) \neq J(p)$ for at least one $p \in \mathbb{P}_{d+1}$.
7. The **truncation error** is defined by $R(f) := I(f) - J(f)$.
8. Recall that the **Lagrange basis** of degree $n-1$ is given by (cf. Chap. 6)
$$\ell_i(x) := \prod_{\substack{j=1 \\ j\neq i}}^n \frac{t - x_j}{x_i - x_j}, \quad i = 1, \ldots, n.$$
9. If $f \in C^n[a, b]$ and $p_n \in \mathbb{P}_{n-1}$ satisfies $p_n(x_i) = f(x_i)$, $i = 1, \ldots, n$ then there is a point ξ_x in the smallest interval containing $[a, b]$ and every x_j such that (cf. Chap. 6)
$$\int_a^b f(x)dx = \int_a^b p_n(x)dx + \frac{1}{n!}\int_a^b f^{(n)}(\xi_x)\Pi(x)\,dx. \tag{9.1}$$

If f_i is a shorthand for $f(x_i)$, with $x_i = a + (i-1)\dfrac{b-a}{n-1}$, we obtain the first Newton-Cotes formulae

n	Name	Formula	Error term
2	Mid-point rule	$(b-a)f_{3/2}$	$\dfrac{(b-a)^3}{24} f^{(2)}(\xi)$
2	Trapezoid rule	$\dfrac{b-a}{2}(f_1+f_2)$	$-\dfrac{(b-a)^3}{12} f^{(2)}(\xi)$
3	Simpson's rule	$\dfrac{b-a}{6}(f_1+4f_2+f_3)$	$-\dfrac{(b-a)^5}{2880} f^{(4)}(\xi)$

Roger Cotes (1682–1716) was an English mathematician, known for working closely with Isaac Newton by proofreading the second edition of his famous book, the Principia. He also invented the quadrature formulas known as Newton-Cotes formulas and first introduced what is known today as Euler's formula.

Thomas Simpson (1710–1761) was a British mathematician, inventor and eponym of Simpson's rule to approximate definite integrals. The attribution, as often in mathematics, can be debated: this rule had been found 100 years earlier by Johannes Kepler. Apparently, the method that is known as Simpson's rule was well known and used earlier by Bonaventura Cavalieri.

The weighted rules are presented in Sect. 9.1.2. ♠

9.1 Basic Quadrature Rules

9.1.1 Quadrature Rules and Degree of Precision

Exercise 9.1. Lagrange polynomial for integration.

Suppose a function $f : [a,b] \to \mathbb{R}$ is only known at certain distinct **sites** $\boldsymbol{x} = [x_1, \ldots, x_n]$, where $x_i \in [a,b]$, for $i = 1, \ldots, n$. Let $p_n \in \mathbb{P}_{n-1}$ be the Lagrange interpolating polynomial at these sites.

1. Show that the basic quadrature $J(f) := \int_a^b p_n(t)\,dt$ satisfies $J(f) = \sum_{j=1}^{n} w_j f(x_j)$ where the w_js depend on the Lagrange basis. ▷*Math Hint*[1]◁
2. Show that $J(f)$ has degree of precision at least $n-1$. ▷*Math Hint*[2]◁
3. Show that if $f \in C^n([a,b])$, then the truncation error can be bounded in terms of the nodal polynomial as follows:

[1] See Exercise 6.3.

[2] Express $q \in \mathbb{P}_{n-1}$ in terms of the Lagrange basis $\{\ell_1, \ldots, \ell_n\}$ to show that $I(q) = J(q)$.

$$|R(f)| \leq \frac{1}{n!} \max_{t\in[a,b]} |f^{(n)}(t)| \int_a^b |\Pi_n(t)|\, dt. \tag{9.2}$$

▷*Math Hint*[3] ◁

Exercise 9.2. Basic quadratures for small n.

We use the notations of the previous exercise with $[a, b] = [-1, 1]$. For each rule find $J(f)$ and use (9.2) to bound the error $R(f) = I(f) - J(f)$.

1. **Mid-point rule** $J_{MP}(f) : n = 1$ and $x_1 = 0$.
2. **Trapezoidal rule** $J_T(f) : n = 2$, $x_1 = -1$ and $x_2 = 1$.
3. **Simpson's rule** $J_S(f) : n = 3$, $x_1 = -1$, $x_2 = 0$ and $x_3 = 1$.
4. Show that $J_S(f) = \dfrac{2J_{MP}(f) + J_T(f)}{3}$.

☺**Comment:** *The corresponding formulas for an arbitrary interval $[a, b]$ are given in the review, with a formula for the truncation error.*☻

Exercise 9.3. Two basic quadrature rules. In the following, for a function $f : [a, b] \to \mathbb{R}$, f_i is a shorthand for $f(x_i)$, with $x_i = a + (i - 1)(b - a)/(n - 1)$.

1. For $n = 4$, consider **Simpson's 3/8 rule**

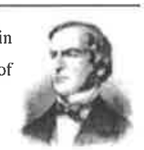
$$J_{S38}(f) = \frac{b-a}{8}(f_1 + 3f_2 + 3f_3 + f_4).$$

Choose the interval $[0, 1]$. Find the exact degree of precision. The error is given by $R_{S38}(f) = c_{S38} f^{(4)}(\xi)$. Find c_{S38} using MATLAB and a polynomial for f.

2. Same questions with **Boole's rule** for $n = 5$.

$$J_B(f) = \frac{b-a}{90}(7f_1 + 32f_2 + 12f_3 + 32f_4 + 7f_5).$$

George Boole (1815–1864) was an English mathematician, philosopher and logician. He worked in the fields of differential equations and algebraic logic and is regarded as a founder of the field of computer science.



Exercise 9.4. Maximum degree of precision.

1. Given two real numbers α_1 and α_2 and two values t_1, t_2 in $[-1, 1]$. For a function $f : [-1, 1] \to \mathbb{R}$, consider the basic quadrature $J_1(f) := \alpha_1 f(t_1) + \alpha_2 f(t_2)$ to approximate $\int_{-1}^{1} f(t)\, dt$. How do we choose the four numbers α_1, α_2, t_1 and t_2 to obtain the maximum degree of precision? ▷*Math Hint*[4]◁

[3] See (9.1).

[4] Write the equations $J_1(p_i) = \int_{-1}^{1} p_i(t)\, dt$ for $p_i(t) = t^i$ and $i = 0, 1, 2, \ldots$

☺**Comment:** *The computed values of the α_i's give the Gauss-Legendre rule (see Exercise 9.6).* ☻

2. Now we consider the quadrature $J_2(f) := \beta_1 f(-1) + \beta_2 f(0) + \beta_3 f(1)$. Compute the values of the β_is to obtain degree of precision 2. Is 2 the maximum degree of precision? ▷*Math Hint*[5]◁

☺**Comment:** *The computed values of the β_i's give the Simpson rule, see the review and Exercise 9.2* ☻

Exercise 9.5. Also in dimension 2.

Given a triangle T in the plane with surface area $A(T)$ and vertices $P_i = (x_i, y_i)$, $i = 1, 2, 3$. Can we choose a point $\hat{P} = (\hat{x}, \hat{y})$ such that

$$\iint\limits_T f(x,y)\,dx\,dy = A(T) f(\hat{P}) \tag{9.3}$$

for any linear function $f(x, y)$ defined on the triangle?

☺**Comment:** *Those who are familiar with triangular barycentric coordinates can go directly to Part 3.* ☻

♠**Review:** If the vertices of T are ordered counterclockwise then $A(T) = \frac{1}{2}\det \boldsymbol{M}_T > 0$, where

$$\boldsymbol{M}_T := \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{bmatrix}. \tag{9.4}$$

The affine function $L(x, y) := [x_1, y_1]^T + \boldsymbol{M}_T[x, y]^T$ maps the unit triangle T_0 with vertices $\{(0,0), (1,0), (0,1)\}$ in a one-to-one fashion onto T. ♠

1. Consider first (9.3) for $T = T_0$ ▷*Math Hint*[6] ◁
2. Treat the general case using the mapping L in the review.
3. Treat the general case using barycentric coordinates (u, v, w) on T.

▷*Math Hint*[7]◁

9.1.2 Weighted Quadrature Rules

♠**Review:**

1. **weighted quadrature rule**:

$$J(f) = \sum_{i=1}^{n} w_i f(x_i) \approx I(f) = \int_a^b f(x) w(x)\,dx,$$

[5] Once the β_is are known, try the quadrature with $p(t) = t^j$ for $j = 3, 4, \ldots$.

[6] In this case we have $\iint_{T_0} f(x,y)\,dx\,dy = \int_0^1 \int_0^{1-y} f(x,y)\,dx\,dy$.

[7] If P_1, P_2, P_3 are the vertices of T, for $u + v + w = 1$ and $u, v, w \geq 0$, then $f(uP_1 + vP_2 + wP_3) = uf(P_1) + vf(P_2) + wf(P_3)$.