Integration

$$S = \int_a^b f(x)dx$$

- ⇨ Exact integration
- ⇨ Simple numerical methods
- ⇨ Advanced numerical methods

Part 1

Exact integration

Three possible ways for exact integration

- Standard techniques of integration substitution rule, integration by parts, using identities, …
- Tables of integrals
- Computer algebra systems

Tables of integrals

Table of Integrals, Series and Products
by Gradshteyn I. S. and Ryzhik I. M.
Academic Press, 1994 (many editions since 195x)
(most referenced in physics)

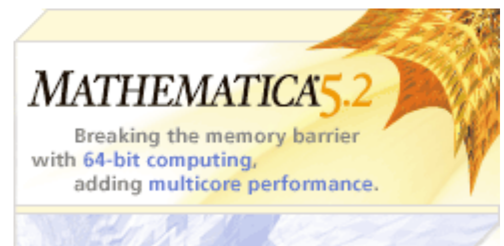
Integral and Series, vol.1-3,
by Prudnikov A P, Brychkov Yu A and Marichev A I
Gordon and Breach, New York, 1986
(most sophisticated)

Tables of Integrals and Other Mathematical Data
by Herbert B. Dwight
(very simple integrals)

and many more …

Computer algebra systems

- Maple
- Mathematica
- MathCad
- Scientific Workplace
- Derive



mathcad 13
driving innovation excellence™

Part 2

Basic ideas

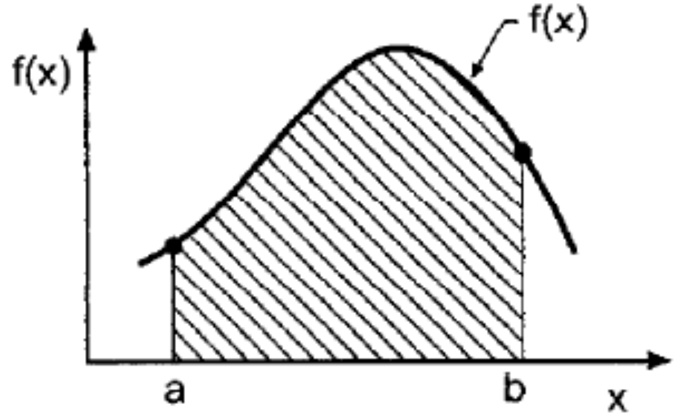
Quite often we need numerical integrations

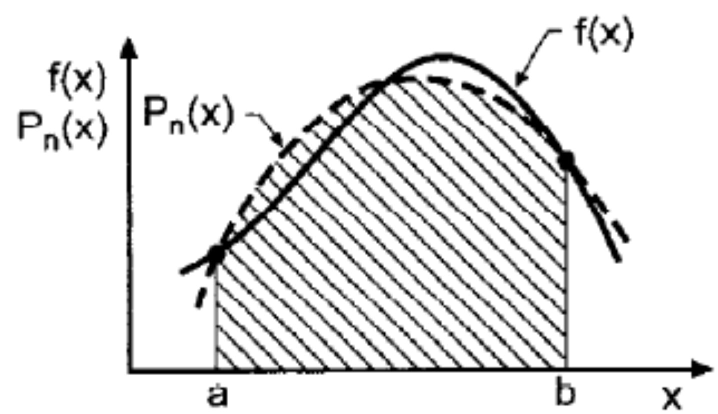
- ⇨ if you can not get an analytic answer using
  Tables of integrals
  Computer algebra systems
  … and various calculus books
- ⇨ if you have a discrete set of data points, i.e. $f_i(x_i)$
  as a result of measurements or calculations

Integrating approximating functions

Numerical integration can be based on fitting approximating functions (polynomials) to discrete data and integrating approximating functions

$$I = \int_a^b f(x)dx \cong \int_a^b P_n(x)dx$$





Integrating approximating functions

Case 1:

The function to be integrated is known only at a finite set of discrete points

Parameters under control – the degree of approximating polynomial

Integrating approximating functions

Case 2:

The function to be integrated is known.

Parameters under control

- ⇨ The total number of discrete points
- ⇨ The degree of the approximating polynomial to represent the discrete data.
- ⇨ The locations of the points at which the known function is discretized

Part 3a

Direct fit polynomials

Direct fit polynomials

The procedure can be used for both unequally and equally spaced data

It is based on fitting the data by a direct fit polynomial and integrating that polynomial.

$$f(x) \cong P_n(x) = a_0 + a_1 x + a_2 x^2 + \dots$$

then

$$I = \int_a^b f(x)dx \cong \int_a^b P_n(x)dx = \left(a_0 x + a_1 \frac{x^2}{2} + a_2 \frac{x^3}{3} + \dots \right)_a^b$$

Part 3b

Quadrature methods on equal subintervals

Riemann Integral

If $f(x)$ is a continuous function defined for $a \le x \le b$ and we divide the interval into n subintervals of equal width $\Delta x = (b-a)/n$ then the definite integral is

$$\int_a^b f(x)dx = \lim_{n\to\infty} \sum_{i=1}^{n} f(x_i^*)\Delta x_i$$

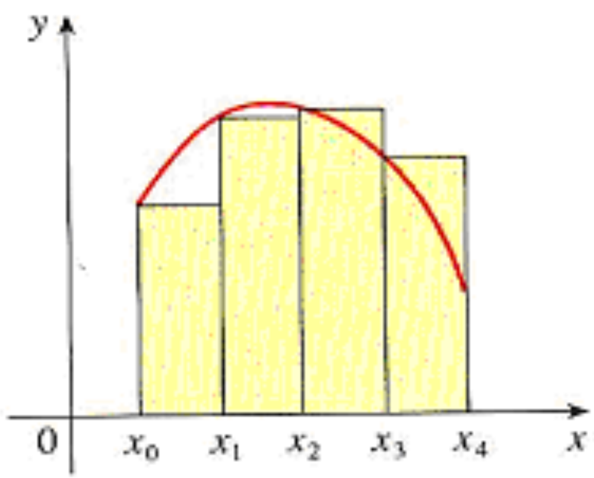
The Riemann integral can be interpreted as a net area under the curve $y = f(x)$ from a to b

Bernhard Riemann, 1826-1866, German mathematician

Primitive rules

1. The left endpoint Riemann sum

$$\int_a^b f(x)dx = L_n = \lim_{n\to\infty}\sum_{i=1}^{n} f(x_{i-1})\Delta x, \quad \Delta x = (b-a)/n$$

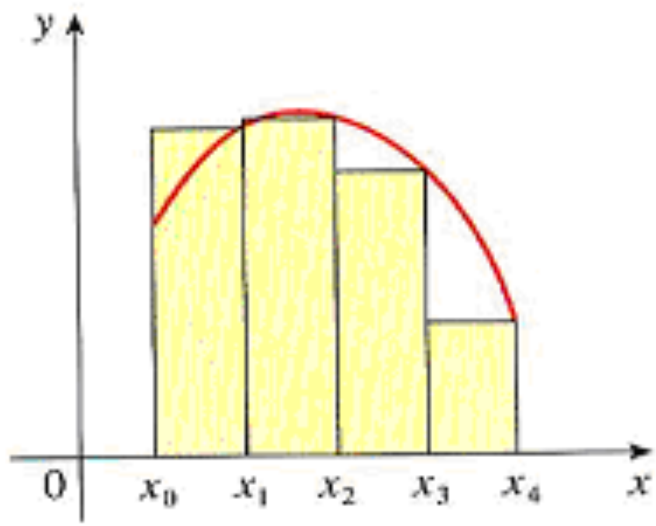


(a) Left endpoint approximation

Primitive rules

2. The right endpoint Riemann sum

$$\int_a^b f(x)dx = R_n = \lim_{n\to\infty}\sum_{i=1}^{n} f(x_i)\Delta x, \quad \Delta x = (b-a)/n$$

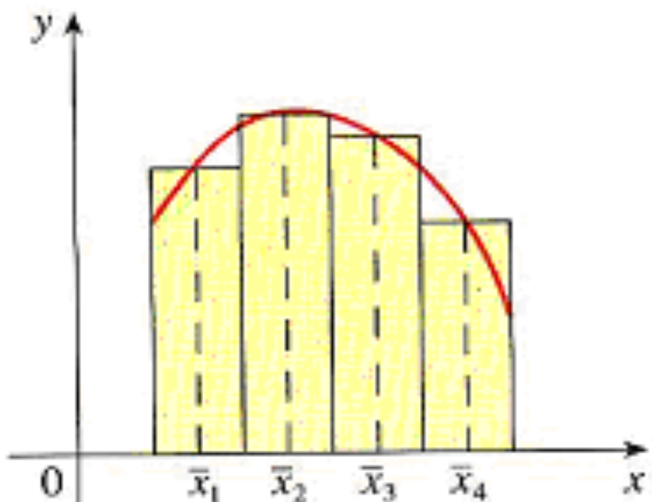


(b) Right endpoint approximation

Primitive rules

3. The midpoint Riemann sum

$$\int_a^b f(x)dx = R_n = \lim_{n\to\infty} \sum_{i=1}^{n} f(\frac{x_{i-1} + x_i}{2})\Delta x, \quad \Delta x = (b-a)/n$$



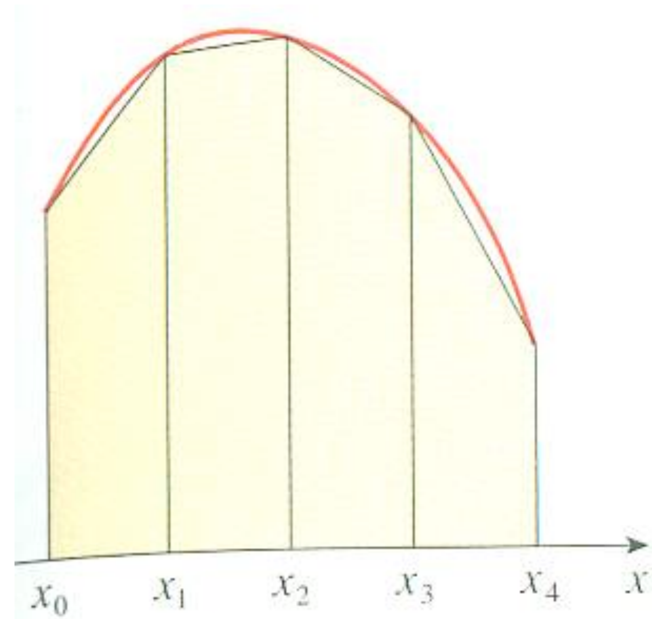
(c) Midpoint approximation

Primitive rules

4. The trapezoidal rule

The area of the trapezoid that lies above the i-th subinterval

$$S_i = \frac{\Delta x}{2}\left(f(x_{i-1}) + f(x_i)\right)$$



$$\int_a^b f(x)dx = \frac{\Delta x}{2}[f(x_0) + 2f(x_1) + 2f(x_2) + \ldots$$
$$2f(x_{n-1}) + f(x_n)]$$

Example: C++

```
// Integration by the trapezoidal rule of f(x) on [a,b]
// f - Function to integrate (supplied by a user)
//    a - Lower limit of integration
//    b - Upper limit of integration
//    r - Result of integration (out)
//    n - number of intervals
double int_trap(double(*f)(double),
                double a, double b, int n)
{
    double r, dx, x;
    r = 0.0;
    dx = (b-a)/n;
    for (int i = 1; i <= n-1; i=i+1)
    {
        x = a + i*dx;
        r = r + f(x);
    }
    r = (r + (f(a)+f(b))/2.0)*dx;
    return r;
}
```

Let us talk about … interpolation

First-order interpolation for the i-th subinterval

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

Integral for the i-th subinterval

$$\int_{x_{i-1}}^{x_i} f(x)dx = \int_{x_{i-1}}^{x_i} \left[f(x_{i-1}) + \frac{f(x_i) - f(x_{i-1})}{\Delta x}(x - x_{i-1}) \right] dx =$$

$$= \frac{\Delta x}{2}\left(f(x_{i-1}) + f(x_i) \right)$$

Trapezoidal approximation is application of the 1st order interpolation for the each subinterval

Integration and the second order interpolation

Using the three-point interpolation one may write

Simpson's Rule for integration

$$\int_a^b f(x)dx = \frac{\Delta x}{3}[f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + \dots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)]$$

Number of n intervals should be even. If n is odd then the last interval should be treated by some other way

Useful exercise: Derive the Simpson rule with a pair of slices with an equal interval by using second-order interpolation for f(x) in the region $[x_{i-1}, x_{i+1}]$.

Thomas Simpson, 1710-1761, England

Example: C++

```
// integration by Simpson rule of f(x) on [a,b]
// f - Function to integrate (supplied by a user)
//    s - Result of integration (out)
//    n - number of intervals

double int_simp(double(*f)(double),
                double a, double b, int n)
{   double s, dx, x;
// if n is odd we add +1 interval to make it even
    if((n/2)*2 != n) {n=n+1;}
    s = 0.0;
    dx = (b-a)/n;
 for ( int i=0; i <n/2; i=i+1)
    { x = a+2*i*dx;
        s = s + 4.0/3.0*f(x); }

 for ( int i=1; i <n/2; i=i+1)
    { x = a+2*i*dx;
        s = s + 2.0/3.0*f(x); }

    s = (s + (f(a) + f(b))/3.0)*dx;
    return s;}
```

Part 3c

Newton-Cotes formulas

Equally spaced points ->
Newton forward difference polynomials

Newton forward/backward/centered difference polynomial are very useful in interpolation and numerical differentiation. Let's consider numerical integration

$$I = \int_a^b f(x)dx \cong \int_a^b P_n(x)dx$$

where

$$P_n(x) = f_0 + s\Delta f_0 + \frac{s(s-1)}{2!}\Delta^2 f_0 + \frac{s(s-1)(s-2)}{3!}\Delta^3 f_0 +$$

$$+\ldots+\frac{s(s-1)(s-2)\ldots(s-[n-1])}{n!}\Delta^n f_0 + \text{Error}$$

$$s = (x-x_0)/h, \quad x = x_0 + sh$$

Comments

using s as a variables

$$I = \int_a^b f(x)dx \cong \int_a^b P_n(x)dx = h\int_{s(a)}^{s(b)} P_n(s)ds = h\int_0^s P_n(x_0 + sh)ds$$

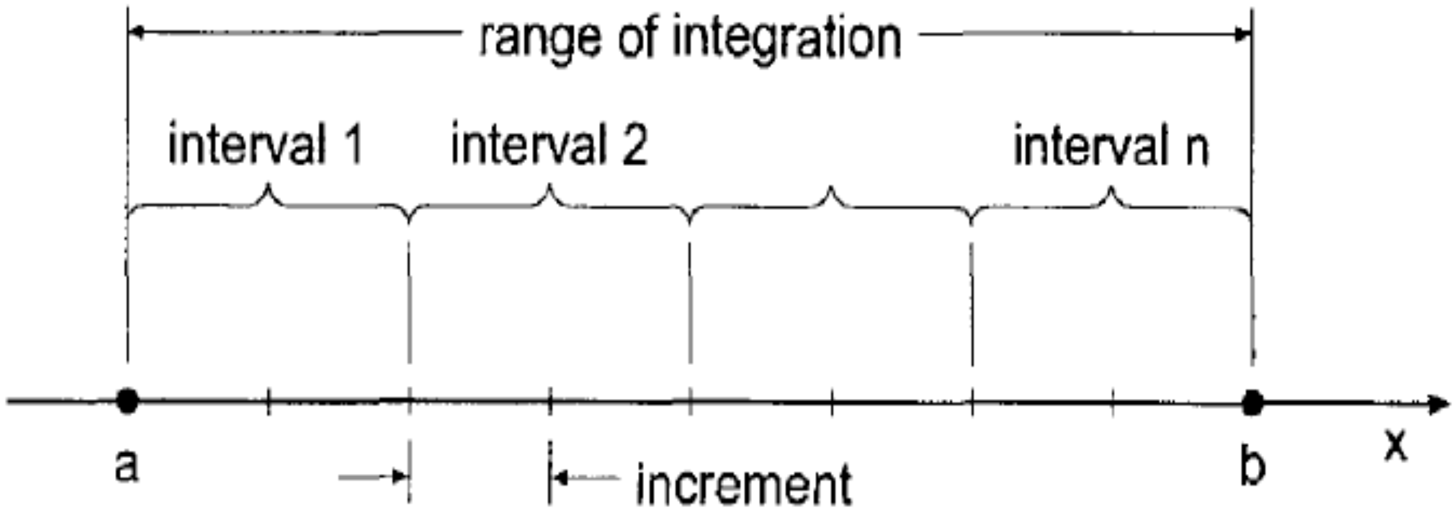
where

$$dx = hds$$

Each choice of the degree n of the interpolating polynomial yields a different Newton-Cotes formula.

See Abramowitz and Stegun (1964) for many high-order formulas

Some terminology



The trapezoid rule (revisited)

A first degree polynomial for a single interval (two points)

$$\Delta I = h\int_0^1 (f_0 + s\Delta f_0)ds = h\left(sf_0 + \frac{s^2}{2}\Delta f_0 \right)\Bigg|_0^1$$

where

$$s = 1, \quad h = \Delta x, \quad \Delta f_0 = (f_1 - f_0)$$

$$\Delta I = h(f_0 + \frac{1}{2}\Delta f_0) = \frac{1}{2}h(f_0 + f_1)$$

applying over all the intervals

$$I = \sum_{i=0}^{n-1} \Delta I_i = \sum_{i=0}^{n-1} \frac{1}{2}h(f_i + f_{i+1}) = \frac{1}{2}h\left(f_0 + 2f_1 + 2f_2 + \ldots + 2f_{n-1} + f_n\right)$$

The trapezoid rule (revisited)

The error estimation can be done by integrating the error term in the polynomial
for a single interval

$$Error = h\int_0^1 \frac{s(s-1)}{2} h^2 f''(\zeta)ds = \frac{1}{12} h^3 f''(\zeta) = O(h^3)$$

the total error

$$Total\ error = \sum_{i=0}^{n-1} Error_i = -\frac{1}{12}(x_n - x_0)h^2 f''(\zeta) = O(h^2)$$

where

$$x_0 \le \zeta \le x_n$$

The Simpson's rule (revisited)

Simpson's 1/3 rule = a second degree polynomial for two intervals (three equally spaced points)

upper limit of integration for a single interval s=2

$$\Delta I = h\int_0^2 \left[f_0 + s\Delta f_0 + \frac{s(s-1)}{2!}\Delta^2 f_0 \right] ds$$

using definitions for $\Delta f_0, \Delta^2 f_0$

$$\Delta I = \frac{1}{3}h(f_0 + 4f_1 + f_2)$$

applying over all the intervals

$$I = \frac{1}{3}h\left(f_0 + 4f_1 + 2f_2 + 4f_3 + \ldots + 4f_{n-1} + f_n\right)$$

The Simpson's rule (revisited)

The error estimation can be done by integrating the error term in the polynomial

for a single interval

$$Error = h\int_0^2 \frac{s(s-1)(s-2)}{6} h^3 f'''(\zeta)ds = 0$$

the next order

$$Error = h\int_0^2 \frac{s(s-1)(s-2)(s-3)}{24} h^4 f''''(\zeta)ds = -\frac{1}{90} h^5 f''''(\zeta)$$

and

$$Total\ error = O(h^4)$$

The Simpson's 3/8 rule

Simpson's 3/8 rule = a third degree polynomial for four equally spaced points

upper limit of integration for a single interval s=3

$$\Delta I = h\int_0^3 \left[f_0 + s\Delta f_0 + \frac{s(s-1)}{2!}\Delta^2 f_0 + \frac{s(s-1)(s-2)}{6}\Delta^3 f \right] ds$$

using definitions for $\Delta f_0, \quad \Delta^2 f_0, \quad \Delta^3 f$

$$\Delta I = \frac{3}{8} h(f_0 + 3f_1 + 3f_2 + f_3)$$

applying over all the intervals

$$I = \frac{3}{8} h\left(f_0 + 3f_1 + 3f_2 + 2f_3 + \ldots + 3f_{n-1} + f_n\right)$$

The Simpson's 3/8 rule

total number of increments must be a multiple of three

Error for a single interval

$$Error = h\int_0^3 \frac{s(s-1)(s-2)(s-3)}{24} h^4 f''''(\zeta)ds = -\frac{3}{80}h^5 f''''(\zeta)$$

and

$$Total\ error = O(h^4)$$

the same order of the error as Simpson's 1/3 rule!

Other view on numerical integration

Quadrature is weighted sum of finite number of sample values of integrand function

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

trapezoid approximations $w_i = (\Delta x / 2, \Delta x / 2)$

Simpson's rule $w_i = (\Delta x / 3, 4\Delta x / 3, \Delta x / 3)$

Equal interval integration (quadrature)

Elementary weights for uniform-step integration rules

name	degree	elementary weights
trapezoid	1	(h/2, h/2)
Simpson's	2	(h/3, 4h/3, h/3)
3/8	3	(3h/8, 9h/8, 9h/8, 3h/8)
Milne	4	(14h/45, 64h/45, 24h/45, 64h/45, 14h/45)

Integration error

Generally as N increases, the precision of a method increases

However, as N increases, the round-off error increases

Some evaluations (not exact but gives an idea)*:

Number of steps for highest accuracy:

trapezoid rule	steps	error
single precision	631	$3*10^{-6}$
double precision	10^{6}	10^{-12}
Simpson's rule	steps	error
single precision	36	$6*10^{-7}$
double precision	2154	$5*10^{-14}$

* see details in R.H. Landau & M.J.Paez, An Introduction to Computational Physics

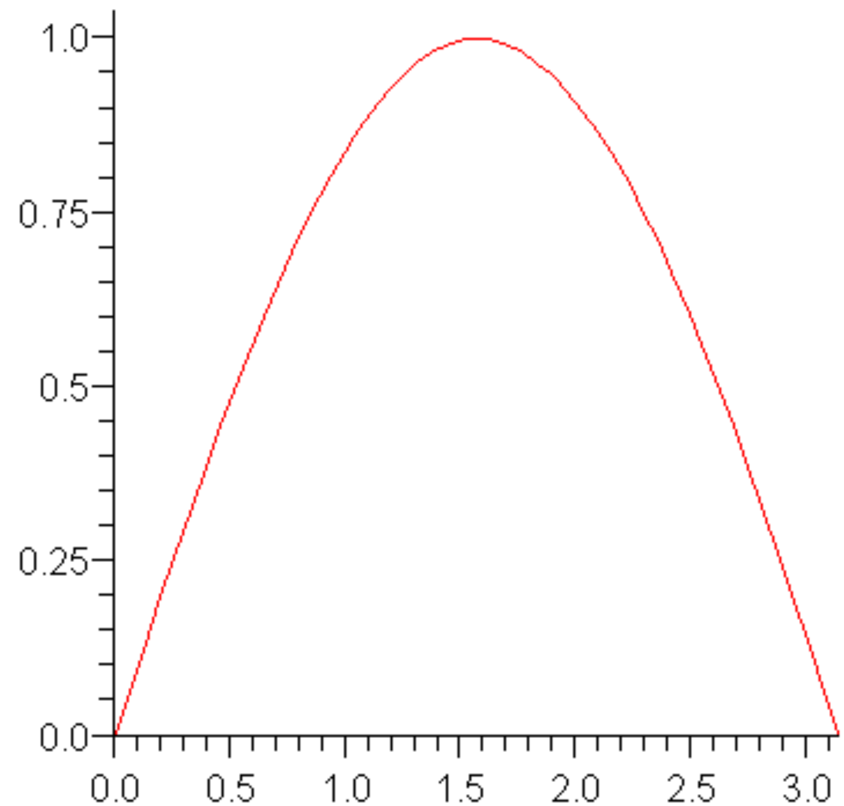
Integration error (cont.)

The best numerical evaluation of an integral can be obtained with a relatively small number is sub-intervals (N~1000) (not with $N \rightarrow \infty$)

It is possible to get an error close to machine precision with Simpson's rule and with other higher-order methods (Newton-Cotes quadrature)

Example $\int_0^{\pi} \sin(x)\,dx = 2.0$

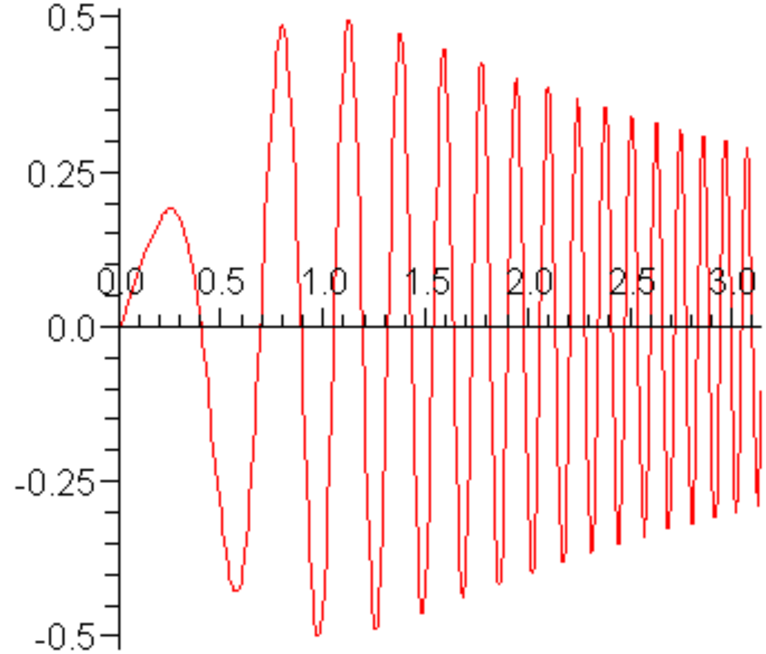
Intervals	Trapez.	Simpson
2	1.570796	2.094395
4	1.896119	2.004560
8	1.974232	2.000269
16	1.993570	2.000017
32	1.998393	2.000001
64	1.999598	2.000000
128	1.999900	2.000000
256	1.999975	2.000000
512	1.999994	2.000000
1024	1.999998	2.000000
2048	2.000000	2.000000




```
   result from quanc8
   nofun =           33
integral =     2.000000
```

Example $\int_0^{\pi} \frac{x}{x^2+1}\cos(10x^2)\,dx = 0.0003156$

```
Intervals   Trapez.    Simpson
        2  0.578769   0.811200
        4  0.813285   0.891458
        8  0.688670   0.647131
       16  0.285919   0.151669
       32  0.049486  -0.029325
       64  0.004360  -0.010682
      128  0.001183   0.000124
      256  0.000526   0.000306
      512  0.000368   0.000315
     1024  0.000329   0.000316
     2048  0.000319   0.000316
     4096  0.000316   0.000316
     8192  0.000316   0.000316
    16384  0.000316   0.000316
    32768  0.000316   0.000316
result from quanc8
    nofun =         1601
 integral =   0.0003156
```



Richardson Extrapolation and Romberg Integration

$$I = \int_a^b f(x)dx$$

Key idea – use the error estimation to extrapolate integrals' values

$$Error(h/R) = \frac{1}{R^n - 1}[I(h/R) - I(h)]$$

where

R is the ratio of the increment size

n is the global order of the algorithm

Extrapolated value = $f(h/R) + Error(h/R)$

When extrapolation is applied to numerical integration by the trapezoid rule, the result is called Romberg integration

Romberg Integration

Error in the trapezoid rule has the functional form

$$Error = C_1 h^2 + C_2 h^4 + C_3 h^6 +$$

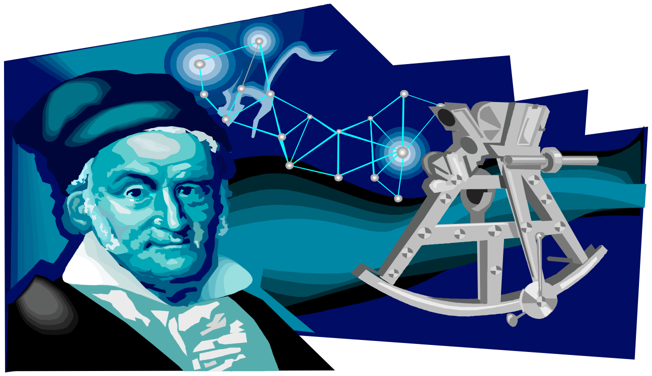
so $n = 2$ and for $R = 2$

$$error(h/2) = \frac{1}{2^n - 1}[I(h/2) - I(h)] = \frac{1}{3}[I(h/2) - I(h)]$$

Extrapolated value = $f(h/2) + error(h/2) + O(h^4)$

Part 4

Gaussian quadrature



Gaussian quadrature

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} c_i f(x_i)$$

Flexibility for known functions – to choose n points x_i and c_i so that the integral of a polynomial of degree 2n-1 is exact.

Gaussian integration produces higher accuracy than the Newton-Cotes formulas with the same number of function evaluations.

If the function to integrate is not smooth, then Gaussian quadrature may give lower accuracy

Example for n=2 and 2n-1=3 $\int_a^b f(x)dx \approx \sum_{i=1}^{n} c_i f(x_i)$

for $f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$

$$\int_a^b f(x)dx \approx c_1 f(x_1) + c_2 f(x_2)$$

$$= c_1\left(a_0 + a_1 x_1 + a_2 x_1^2 + a_3 x_1^3\right) + c_2\left(a_0 + a_1 x_2 + a_2 x_2^2 + a_3 x_2^3\right)$$

$$\int_a^b f(x)dx = \int_a^b \left(a_0 + a_1 x + a_2 x^2 + a_3 x^3\right)dx$$

$$= \left[a_0 x + a_1 \frac{x^2}{2} + a_2 \frac{x^3}{3} + a_3 \frac{x^4}{4}\right]_a^b$$

$$= a_0(b-a) + a_1\left(\frac{b^2 - a^2}{2}\right) + a_2\left(\frac{b^3 - a^3}{3}\right) + a_3\left(\frac{b^4 - a^4}{4}\right)$$

Example for n=2

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} c_i f(x_i)$$

and

$$c_1\left(a_0 + a_1x_1 + a_2x_1^2 + a_3x_1^3\right) + c_2\left(a_0 + a_1x_2 + a_2x_2^2 + a_3x_2^3\right)$$

$$= a_0\left(c_1 + c_2\right) + a_1\left(c_1x_1 + c_2x_2\right) + a_2\left(c_1x_1^2 + c_2x_2^2\right) + a_3\left(c_1x_1^3 + c_2x_2^3\right)$$

$$= a_0(b-a) + a_1\left(\frac{b^2-a^2}{2}\right) + a_2\left(\frac{b^3-a^3}{3}\right) + a_3\left(\frac{b^4-a^4}{4}\right)$$

$$b - a = c_1 + c_2$$

$$\frac{b^2-a^2}{2} = c_1x_1 + c_2x_2$$

$$\frac{b^3-a^3}{3} = c_1x_1^2 + c_2x_2^2$$

$$\frac{b^4-a^4}{4} = c_1x_1^3 + c_2x_2^3$$

four unknowns
and
four equations

Example for n=2

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} c_i f(x_i)$$

and after some tedious work …

$$c_1 = \frac{b-a}{2}$$

$$c_2 = \frac{b-a}{2}$$

$$x_1 = \left(\frac{b-a}{2}\right)\left(-\frac{1}{\sqrt{3}}\right) + \frac{b+a}{2}$$

$$x_2 = \left(\frac{b-a}{2}\right)\left(+\frac{1}{\sqrt{3}}\right) + \frac{b+a}{2}$$

Another way. Choose t_1, t_2, c_1, c_2 so that I is exact for the following four polynomials: $f(t) = 1, t, t^2, t^3$ *(and use a=-1, b=1)*

$$I[F(t) = 1] = \int_{-1}^{1} (1)\, dt = t\Big|_{-1}^{1} = 2 = C_1(1) + C_2(1) = C_1 + C_2$$

$$I[F(t) = t] = \int_{-1}^{1} t\, dt = \tfrac{1}{2}t^2\Big|_{-1}^{1} = 0 = C_1 t_1 + C_2 t_2$$

$$I[F(t) = t^2] = \int_{-1}^{1} t^2\, dt = \tfrac{1}{3}t^3\Big|_{-1}^{1} = \tfrac{2}{3} = C_1 t_1^2 + C_2 t_2^2$$

$$I[F(t) = t^3] = \int_{-1}^{1} t^3\, dt = \tfrac{1}{4}t^4\Big|_{-1}^{1} = 0 = C_1 t_1^3 + C_2 t_2^3$$

Solving the system gives

$C_1 = C_2 = 1$ and $t_1 = -\dfrac{1}{\sqrt{3}}, \quad t_2 = \dfrac{1}{\sqrt{3}}$

$$I = \int_{-1}^{1} F(t)dt = F(-\frac{1}{\sqrt{3}}) + F(\frac{1}{\sqrt{3}})$$

Gaussian quadratures for $I = \int_a^b f(x)dx$

transformation between x and t space

$$x = mt + c$$

where $x = a \rightarrow t = -1, \quad x = b \rightarrow t = 1$ and $dx = m\,dt$

then $a = m(-1) + c$ and $b = m(1) + c$

$$m = \frac{b-a}{2}, \quad c = \frac{b+a}{2} \quad \text{and} \quad x = \frac{b-a}{2}t + \frac{b+a}{2}$$

$$I = \int_a^b f(x)df = \int_{-1}^{1} f[x(t)]dt = \int_{-1}^{1} f(mt+c)mdt$$

$$F(t) = f[x(t)] = f(mt+c)$$

$$I = \frac{b-a}{2}\int_{-1}^{1} F(t)dt = \frac{b-a}{2}\sum_{i=1}^{n} C_i F(t_i)$$

Example: Gaussian quadrature parameters

n	t_i	C_i
2	$-1/\sqrt{3}$	1
	$1/\sqrt{3}$	1
3	$-\sqrt{0.6}$	5/9
	0	8/9
	$\sqrt{0.6}$	5/9
4	−0.8611363116	0.3478548451
	−0.3399810436	0.6521451549
	0.3399810436	0.6521451549
	0.8611363116	0.3478548451

Example: C++

```
/*  Numerical integration of f(x) on [a,b]
    method: Gauss (4 points)
input:
    f   - a single argument real function
    a,b - the two end-points (interval of integration)
output:  r - result of integration
*/
  double gauss4(double(*f)(double), double a, double b)
{
    const int n = 4;
    double ti[n] = {-0.8611363116, -0.3399810436,
                     0.3399810436,  0.8611363116};
    double ci[n] = { 0.3478548451,  0.6521451549,
                     0.6521451549,  0.3478548451};
    double r, m, c;
    r = 0.0;
    m = (b-a)/2.0;
    c = (b+a)/2.0;
    for (int i = 1; i <= n; i=i+1)
    {r = r + ci[i-1]*f(m*ti[i-1] + c); }
    r = r*m;
    return r;
}
```

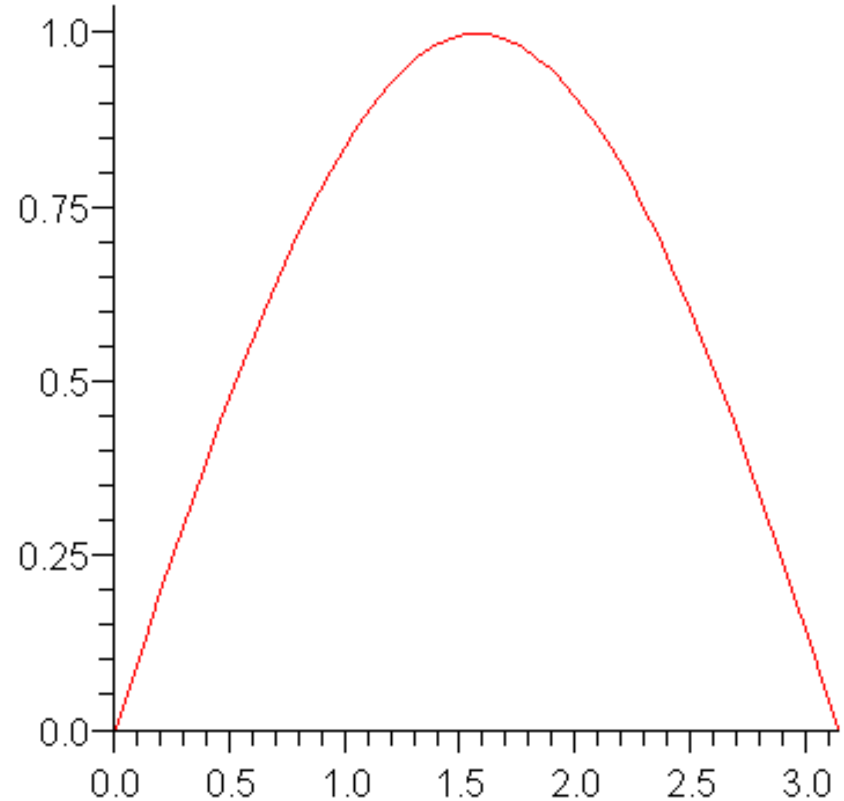
Example: C++

```
/*  Numerical integration of f(x) on [a,b]
    method: Gauss (8 points using symmetry)
input:
    f   - a single argument real function
    a,b - the two end-points (interval of integration)
output:  r - result of integration */
  double gauss8(double(*f)(double), double a, double b)
{
    const int n = 4;
    double ti[n] = {0.1834346424, 0.5255324099,
                    0.7966664774, 0.9602898564};
    double ci[n] = {0.3626837833, 0.3137066458,
                    0.2223810344, 0.1012285362};
    double r, m, c;
    r = 0.0;
    m = (b-a)/2.0;
    c = (b+a)/2.0;
    for (int i = 1; i <= n; i=i+1)
    {r=r+ci[i-1]*(f(m*(-1.0)*ti[i-1]+c)+f(m*ti[i-1]+c));
    }
    r = r*m;
    return r;
}
```

Example $\int_0^{\pi} \sin(x)\, dx = 2.0$

Intervals	Trapez.	Simpson
2	1.570796	2.094395
4	1.896119	2.004560
8	1.974232	2.000269
16	1.993570	2.000017
32	1.998393	2.000001
64	1.999598	2.000000
128	1.999900	2.000000
256	1.999975	2.000000
512	1.999994	2.000000
1024	1.999998	2.000000
2048	2.000000	2.000000

```
gauss4 = 1.999984
gauss8 = 2.000000
```



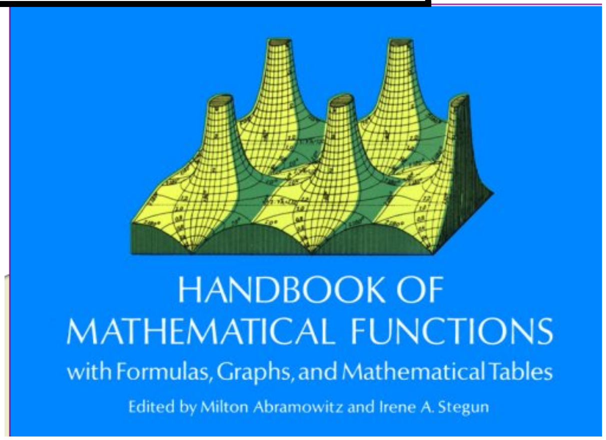

```
   result from quanc8
   nofun =          33
integral =    2.000000
```

Gaussian quadratures

$$\int_a^b w(x) f(x) dx$$

		$w(x)$
Gauss-Legendre	[-1, +1]	1
Gauss-Jacobi	(-1,+1)	$(1-x)^{\alpha}(1+x)^{\beta}$
Gauss-Chebyshev	(-1,+1)	$1/\sqrt{1-x^2}$
Gauss-Hermite	(-∞, +∞)	$\exp(-x^2)$
Gauss-Laguerre	[0, +∞)	$\exp(-x)$

Tables with coefficients can be found in "Handbook of Mathematical Functions, With Formulas, Graphs, and Mathematical Tables" by Abramowitz and Stegun.

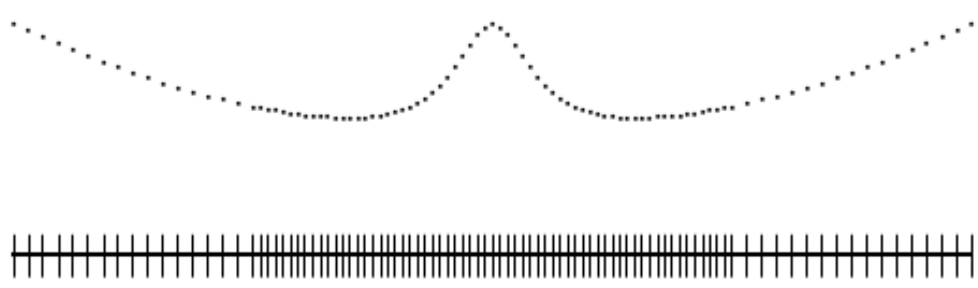


The aim of an automatic integration scheme is to relieve the person who has to compute an integral of any need to think.

Davis P. J., and P. Rabinowitz, Methods of Numerical Integration (Dover, 2nd edition) (2007)

Part 5

Automatic and Adaptive Integration



Automatic and Adaptive Integration

A favorite research topic since the 1960s.

User-friendly routines where the user enters
1) the limits of integration, 2) selects the routine for computation of f(x), 3) provides a tolerance ε, and 4) enters the upper bound N for the number of functional computations.

$$\left| I_a^b(f) - I \right| \le \varepsilon \quad \text{or} \quad \frac{\left| I_a^b(f) - I \right|}{I_a^b(|f|)} \le \varepsilon$$

Then the program exits either
1) with the computed value which is correct within the ε
2) with a statement that the upper bound N was attained but the tolerance was not achieved, and the computed result may be the "best“ value of the integral determined by the program.

Automatic and Adaptive Integration

Automatic integration falls into two classes: iterative or non-iterative, and adaptive or non-adaptive.

The iterative schemes: computing successive approximations to the integral until an agreement with the tolerance is achieved,

The non-iterative schemes: the information from the first approximation is carried over to generate the second approximation, which then becomes the final result.

Adaptive Integration

The adaptive schemes: the points at which the integration is carried out are chosen in a manner that is dependent on the nature of the integrand – the domain of integration is selectively refined to reflect behavior of particular integrand function on a specific subinterval.

The non-adaptive schemes: the integration points are chosen in a fixed manner which is independent of the nature of the integrand, although the number of these points depends on the integrand - continue to subdivide all subintervals, say by half, until overall error estimate falls below desired tolerance (not an inefficient way).

Adaptive programs tend to be effective in practice … but it can be fooled

Interval of integration may be very wide but "interesting" behavior of integrand is confined to narrow range

Sampling by automatic routine may miss interesting part of integrand behavior, and resulting value for integral may be completely wrong

Part 6

"Special cases"

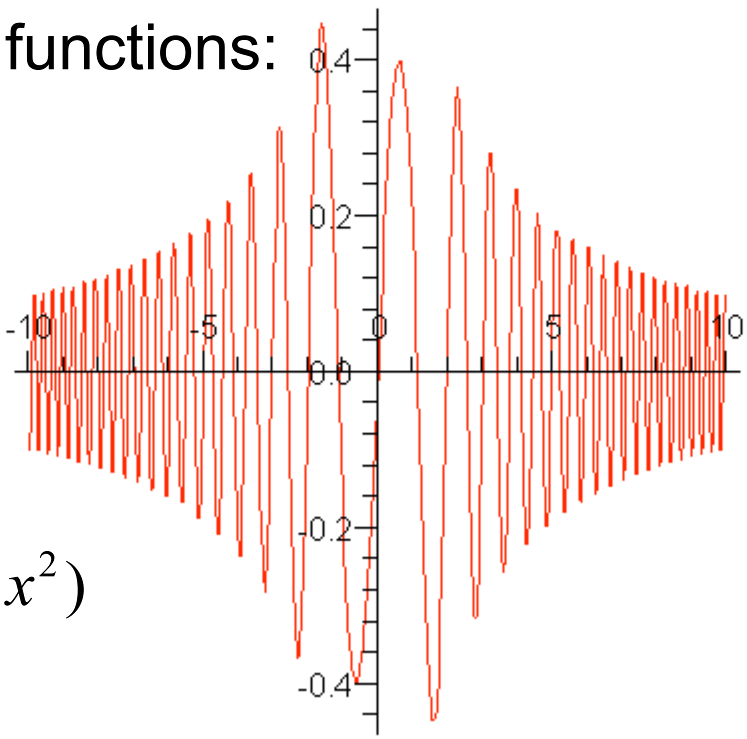
Integrals with oscillating functions

$$\int_a^b f(x)\cos^n(\omega x)\,dx$$

Use methods or programs specially designed to calculate integrals with oscillating functions:

Filon's method

Clenshaw-Curtis method



$$\frac{x}{x^2+1}\cos(1.2x^2)$$

Improper Integrals:
Type 1 - Infinite Intervals

$$\int_a^{\infty} f(x)dx \qquad \int_{-\infty}^{\infty} f(x)dx$$

1. Transform variable of integration so that the new interval is finite: examples: y=exp(-x), then [0,∞] into [0,1] (but: not to introduce singularities)
2. Replace infinite limits of integration by carefully chosen finite values.
3. Use asymptotic behavior (if possible) to evaluate the “tail” contribution.
4. Use nonlinear quadrature rules designed for infinite intervals

example: replace infinite limits of integration by finite values

$$\int_0^\infty f(x)dx = \lim_{r\to\infty} \int_0^r f(x)dx$$

$$I_n = \int_0^{r_n} f(x)dx \quad \text{where } r_n = 2^n$$

$$I_n = \int_0^{r_n} \frac{e^{-x}}{1+x^4}dx, \qquad r_n = 2^n.$$

n	I_n	Number of functional evaluations
0	.5720 2582	35
1	.6274 5952	52
2	.6304 3990	100
3	.6304 7761	178
4	.6304 7766	322
Exact	.6304 7783	

example: using asymptotic behavior

$$\int_0^\infty \frac{\sqrt{x}}{x^2+1}dx = \int_0^a \frac{\sqrt{x}}{x^2+1}dx + \int_a^\infty \frac{\sqrt{x}}{x^2+1}dx$$

for a >> 1 we use the asymptotic behavior of the function

$$\int_a^\infty \frac{\sqrt{x}}{x^2+1}dx \approx \int_a^\infty \frac{\sqrt{x}}{x^2}dx \to \int_a^\infty \frac{1}{x^{3/2}}dx \to \frac{2}{\sqrt{a}}$$

$$\int_0^\infty \frac{\sqrt{x}}{x^2+1}dx \approx \int_0^a \frac{\sqrt{x}}{x^2+1}dx + \frac{2}{\sqrt{a}}$$

exact value

$$\int_0^\infty \frac{\sqrt{x}}{x^2+1}dx = \pi\frac{\sqrt{2}}{2}$$

Example $\int_0^\infty \frac{\sqrt{x}}{x^2+1}dx = \pi \frac{\sqrt{2}}{2} = 2.221441469$

upper limit = 100

no "tail"

Intervals	Trapez.	Simpson
2	0.166362	0.205151
4	0.321345	0.373006
8	0.536673	0.608449
16	0.833630	0.932615
32	1.218034	1.346168
64	1.619001	1.752657
128	1.873848	1.958797
256	1.970354	2.002522
512	2.003473	2.014513
1024	2.015099	2.018974
2048	2.019202	2.020570
4096	2.020652	2.021136
8192	2.021165	2.021336
16384	2.021346	2.021407
32768	2.021410	2.021432

```
result from quanc8
    nofun =          769
 integral =    2.021445
```

with the "tail" = 0.20000

Intervals	Trapez.	Simpson
2	0.366362	0.405151
4	0.521345	0.573006
8	0.736673	0.808449
16	1.033630	1.132615
32	1.418034	1.546168
64	1.819001	1.952657
128	2.073848	2.158797
256	2.170354	2.202522
512	2.203473	2.214513
1024	2.215099	2.218974
2048	2.219202	2.220570
4096	2.220652	2.221136
8192	2.221165	2.221336
16384	2.221346	2.221407
32768	2.221410	2.221432

```
result from quanc8
    nofun =          769
 integral =    2.221445
```

Improper Integrals:
Type 2 - Discontinuous Integrands

$\int_0^1 f(x)dx$ when $f(x)$ is discontinuous at 0

Formal definition

$$\int_0^1 f(x)dx = \lim_{t \to 0+} \int_t^1 f(x)dx$$

Proceeding to the limit $\int_0^1 f(x)dx = \int_{R_1}^1 f(x)dx + \int_{R_2}^{R_1} f(x)dx + \ldots$

Change variables $R_n = 2^{-n}$

Elimination of the singularity

Gauss type quadratures

…

Improper integrals 3:
Integrals with integrable singularity

$$\int_a^b \frac{f(x)}{x-c}dx \qquad a \le c \le b$$

Method 1:

$$f(x) = \varphi(x) + \psi(x)$$

where $\varphi(x)$ can be integrated numerically

and $\psi(x)$ can be integrated analytically

example: $f(x) = \dfrac{1}{\sqrt{x(1+x^2)}}$ (problem at $x = 0$)

$$f(x) = \left(\frac{1}{\sqrt{x(1+x^2)}} - \frac{1}{\sqrt{x}} \right) + \frac{1}{\sqrt{x}}$$

Improper integrals 3:
Integrals with integrable singularity

$$\int_a^b \frac{f(x)}{x-c}\,dx \qquad a \le c \le b$$

Method 2:

$$f(x) = \varphi(x) \cdot \rho(x)$$

then for some cases one of following quadrature rules can be used:

Gauss-Christoffel

Jacoby,

Chebyshev

Improper integrals 3:
Integrals with integrable singularity

$$\int_a^b \frac{f(x)}{x-c}\,dx \qquad a \le c \le b$$

Method 3:

using non-standard quadrature rules allowing explicitly for the singularity

Method 4:

Use programs from trusted numerical libraries or books.

Principal value integrals

$$\int_0^\infty \frac{f(x)}{x - x_0 \pm i\varepsilon} dx \quad \text{where } \varepsilon \to 0$$

$$\lim_{\varepsilon \to 0} \int_0^\infty \frac{f(x)}{x - x_0 \pm i\varepsilon} dx = P\int_0^\infty \frac{f(x)}{x - x_0} dx \mp i\pi f(x_0)$$

the definition of the principal value integral

$$P\int_0^\infty \frac{f(x)}{x - x_0} dx = \lim_{\mu \to 0} \left[\int_0^{x_0 - \mu} \frac{f(x)}{x - x_0} dx + \int_{x_0 + \mu}^\infty \frac{f(x)}{x - x_0} dx \right]$$

it is possible to calculate it using regular methods,

but... there is another way

Principal value integrals (part 2)

$$\int_0^{\infty} \frac{f(x)}{x-x_0}\,dx \approx \int_0^{R} \frac{f(x)}{x-x_0}\,dx$$

$$\int_0^{R} \frac{f(x)}{x-x_0}\,dx = \int_0^{R} \frac{[f(x)-f(x_0)]f(x)}{x-x_0}\,dx + f(x_0)\int_0^{R} \frac{1}{x-x_0}\,dx$$

$$\ln\frac{R-x_0}{-x_0} = \left[\ln(-1) + \ln\frac{R-x_0}{x_0}\right] = \pm i\pi + \ln\frac{R-x_0}{x_0}$$

$$\int_0^{R} \frac{f(x)}{x-x_0 \mp i\varepsilon}\,dx = \int_0^{R} \frac{[f(x)-f(x_0)]f(x)}{x-x_0}\,dx + f(x_0)\ln\frac{R-x_0}{x_0} \pm i\pi f(x_0)$$

then regular methods can be used since

$$\frac{[f(x)-f(x_0)]f(x)}{x-x_0}$$ is smooth around x_0

Double and multiple integrals

Use automatic one-dimensional quadrature routine for each dimension, one for outer integral and another for inner integral

Monte-Carlo method
(effective for large dimensions)

$$\int_0^1 dx_1 \int_0^1 dx_2 \int_0^1 dx_3 \int_0^1 dx_4 \int_0^1 dx_5 \int_0^1 dx_6 \int_0^1 (x_1 + x_2 + \ldots + x_7)^2 \, dx_7$$

Integrating Tabular Data

Reasonable approach is to integrate piecewise interpolant

Cubic spline interpolation could be a good method.


HANDBOOK OF
Computational Methods for Integration
Prem K. Kythe
Michael R. Schäferkotter
CHAPMAN & HALL/CRC
PHILIP J. DAVIS AND
PHILIP RABINOWITZ
Methods of Numerical Integration
SECOND EDITION
Lecture Notes in Computer Science
848
Arnold R. Krommer
Christoph W. Ueberhuber
Numerical Integration
on Advanced Computer Systems
Springer-Verlag
Approximate Calculation of Integrals
V. I. Krylov
Translated by Arthur H. Stroud

too sad

95 % of all practical work in numerical analysis boiled down to applications of Simpson's rule and linear interpolation.

Milton Abramowitz

from Davis P. J., and P. Rabinowitz, Methods of Numerical Integration (Dover, 2nd edition) (2007)

Conclusion

$$S = \int_a^b f(x)dx$$

- Analyze first: the existence of the integral
- Transform the integral to a simpler form (if possible)
- Analyze the function: smooth or oscillating, functions with singularities, narrow peaks, …
- Analyze to type of the integral (regular, improper, …)
- Select a method that fits the function and the integral
- **Always** test any program for integration before using for real calculations.