Numerical Methods I: Numerical Integration/Quadrature

Georg Stadler
Courant Institute, NYU
stadler@cims.nyu.edu

November 30, 2017

Numerical integration

We want to approximate the definite integral

$$I(f) = I_a^b(f) = \int_a^b f(t)\,dt$$

numerically. Properties of the Riemann integral:

- I is linear
- positive, i.e., if f is nonnegative, then $I(f)$ is nonnegative.
- additive w.r.t. to the interval bounds: $I_a^c = I_a^b + I_b^c$

Numerical integration
Conditioning

$\|f\|_1 = \int_a^b |f(t)|\,dt$ "1-norm"

Lets study the map

$$([a,b],f) \to \int_a^b f(t)\,dt,$$

where we use the L^1-norm for f. The absolute and relative condition numbers of integration are:

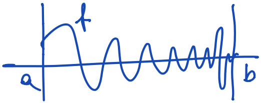
recall $(a,b) \longmapsto a+b$

$a,b \in \mathbb{R}$

$\kappa_{rel} = \frac{|a|+|b|}{|a+b|}$

$$\kappa_{\text{abs}} = 1,$$
$$\kappa_{\text{rel}} = \frac{I(|f|)}{I(f)}.$$

So, integration is harmless w.r. to the absolute condition number, and problematic w.r. to the relative condition number if $I(f)$ is small and f changes sign.



poorly conditioned for f that change sign often.

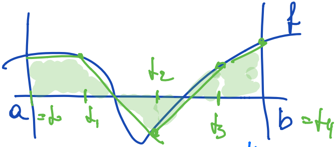
Numerical integration

We are looking for a map

$$\hat{I} : \begin{cases} C([a,b]) & \to \mathbb{R} \\ f & \mapsto \hat{I}(f) \end{cases}$$

such that $|I(f) - \hat{I}(f)|$ is small.

Example: Trapezoidal rule.



$$\hat{I}(f) = \sum_{i=1}^{4} (f(t_{i-1}) + f(t_i)) \cdot \frac{(t_i - t_{i-1})}{2}$$

General quadrature formula:

$$\hat{I}(f) = \sum_{i=0}^{n} \lambda_i f(t_i),$$

with weights λ_i and nodal points t_i, $i = 0, 1, \ldots, n$.

Replace f by easy-to-integrate approximation $\hat{f}$, and set

$$\hat{I}(f) := I(\hat{f}).$$

Given fixed nodes $t_0, \ldots, t_n$, use polynomial approximation

$$\hat{f} = P(f|t_0, \ldots, t_n) = \sum_{i=0}^{n} f(t_i) L_{in}(t)$$

Lagrange polynomials

$$L_{in}(t) = \prod_{k \neq i} \frac{(t - t_k)}{(t_i - t_k)}$$

Thus:

$$\hat{I}(f) = (b-a) \sum_{i=0}^{n} \lambda_{in} f(t_i),$$

where $\lambda_{in} = \frac{1}{b-a} \int_a^b L_{in}(t)\, dt$

Numerical integration

Newton-Cotes formulas

Theorem: For $(n+1)$ pairwise distinct points, there exists exactly one quadrature formula that is exact for all $p \in \boldsymbol{P}_n$.

Uniformly spaced points:

$$\lambda_{in} = \frac{1}{b-a} \int_a^b \prod_{i \neq j} \frac{t - t_i}{t_i - t_j} \, dt = \frac{1}{n} \int_0^n \prod_{i \neq j} \frac{s - j}{i - j} \, ds$$

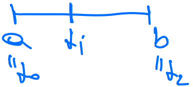
These weights are independent of a and b.

$\hat{I}(f) = \frac{1}{b-a}\left(\frac{f(a)}{2} + \frac{f(b)}{2}\right)$

Table 9.1. Newton-Cotes weights λ_{in} for $n = 1, \ldots, 4$.

$h = b-a$

n	$\lambda_{0n}, \ldots, \lambda_{nn}$	Error	Name
1	$\frac{1}{2}$ $\frac{1}{2}$	$\frac{h^3}{12} f''(\tau)$	Trapezoidal rule
2	$\frac{1}{6}$ $\frac{4}{6}$ $\frac{1}{6}$	$\frac{h^5}{90} f^{(4)}(\tau)$	Simpson's rule, Kepler's barrel rule
3	$\frac{1}{8}$ $\frac{3}{8}$ $\frac{3}{8}$ $\frac{1}{8}$	$\frac{3h^5}{80} f^{(4)}(\tau)$	Newton's 3/8-rule
4	$\frac{7}{90}$ $\frac{32}{90}$ $\frac{12}{90}$ $\frac{32}{90}$ $\frac{7}{90}$	$\frac{8h^7}{945} f^{(6)}(\tau)$	Milne's rule



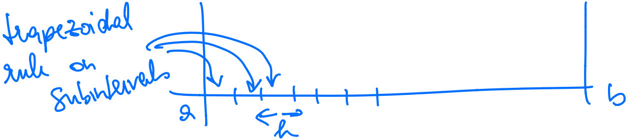
Weights are positive up to order 7, then some become negative.

Proof of error terms is based on

- a variant of the mean value theorem for integrals
- the Newton remainder form from interpolation, e.g., for linear interpolation

$$f(t) - P(t) = [t, a, b]f \cdot (t-a)(t-b) = \frac{f''(\tau)}{2}(t-a)(t-b).$$

For piece-wise polynomial approximations (e.g., the trapezoidal rule), analogous estimates as shown in the table hold.



quadrature rule

$$\hat{I}(f) = \sum_{i=0}^{n} \lambda_i f(t_i)$$

weights nodes

Now, let's allow the nodes $t_0, \ldots, t_n$ to vary. The best we can hope for is exact interpolation up to polynomials of degree $2n+1$ (based on a non-rigorous counting argument).

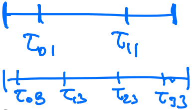
Also, for generalization, we consider quadrature of weighted integrals, with a positive weight function $\omega(t)$:

$$I(f) = \int_a^b \omega(t) f(t)\, dt$$

with weight funcations $\omega(t) = 1, \omega(t) = 1/\sqrt{1-t^2}, \ldots$.

$j=1$:

$j=3$



Theorem: If $\hat{I}$ is exact for polynomials $p \in \boldsymbol{P}_{2n+1}$ (for ω-weighted integration), then

quadrature nodes

$$P_{j+1}(t) = (t-\tau_{0j}) \cdot \ldots \cdot (t-\tau_{jj}) \in \boldsymbol{P}_{n+1}$$

are orthogonal with respect to the scalar product induced by $\omega(t)$.

Proof: $P_{n+1}P_j$, $j \le n$ is in $\boldsymbol{P}_{2n+1}$, $\hat{I}$ is exact $\Longrightarrow$

$$\hat{I}(P_{n+1}P_j) = \int_a^b \omega(t) P_{n+1}(t) P_j(t)\, dt = (P_{n+1}, P_j)$$

exact

$$= \sum_{i=0}^{n} \lambda_{in} \underbrace{P_{n+1}(\tau_{in})}_{=0} P_j(\tau_{in}) = 0$$

$\Longrightarrow (P_{n+1}, P_j) = 0$ $j = 0, \ldots, n$ i.e. $P_{n+1} \perp P_j$

$\Longrightarrow P_{n+1}$ is orthogonal polynomial. ✓

Thus:

- one must choose the roots of the orthogonal polynomials (which are single roots)
- the weights are uniquely determined and yield exact integration for polynomials up to degree n... but:

Theorem: Let $\tau_{0n}, \ldots, \tau_{nn}$ be the roots of the $(n+1)$st orthogonal polynomial for the weight ω. Then any quadrature formula $\hat{I}$ is exact for polynomials up to order n if and only if it is exact up to order $2n+1$.

Numerical integration

Gauss-Christoffel quadrature

"$\Leftarrow$" ✓

"$\Rightarrow$": nodes are the roots of P_{n+1}, integration is exact up to degree n. Have to show that $\hat{I}$ is exact up to degree $2n+1$.

$P \in \mathbb{P}_{2n+1}$, $P = Q P_{n+1} + R$, $Q, R \in \mathbb{P}_n$

exact for deg $\leq n$

$$\int_a^b w(t) P(t)\, dt = \int_a^b w(t) \big(\underbrace{Q P_{n+1}}_{=0} + R\big)(t)\, dt = \int_a^b w(t) R(t)\, dt = \hat{I}(R)$$

$= 0$ because $P_{n+1} \perp$ all polynomials of degree n or less

$$\hat{I}(R) = \sum_{i=0}^{n} \lambda_{in} R(\tau_{in}) = \sum_{i=0}^{n} \lambda_{in} \Big[R(\tau_{in}) + Q(\tau_{in}) \underbrace{P_{n+1}(\tau_{in})}_{=0} \Big]$$

$$= \hat{I}(P)$$

$\square$ ✓

$\omega(t)$	Interval $I = [a, b]$	Orthogonal polynomials
$\frac{1}{\sqrt{1-t^2}}$	$[-1, 1]$	Chebyshev polynomials T_n
e^{-t}	$[0, \infty]$	Laguerre polynomials L_n
e^{-t^2}	$[-\infty, \infty]$	Hermite polynomials H_n
1	$[-1, 1]$	Legendre polynomials P_n

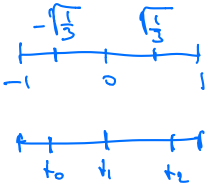
Corresponding quadrature rules are usually prefixed with "Gauss-", i.e., "Gauss-Legendre quadrature", or "Gauss-Chebyshev quadrature".

Numerical integration

Gauss-Legendre points/weights for interval $[-1, 1]$

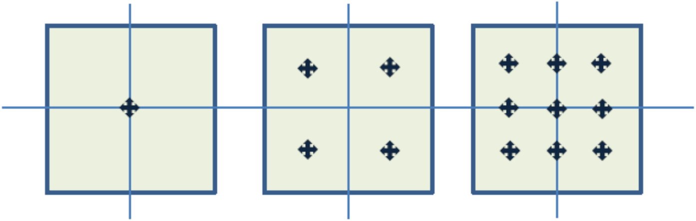
$$\int \overbrace{x^3 + x^2 - 1}^{f}\, dx = f\left(-\sqrt{\tfrac{1}{3}}\right) + f\left(\sqrt{\tfrac{1}{3}}\right)$$

Number of points, *n*	Points, x_i	Weights, w_i
1	0	2
2	$\pm\sqrt{\frac{1}{3}}$	1
3	0	$\frac{8}{9}$
	$\pm\sqrt{\frac{3}{5}}$	$\frac{5}{9}$
4	$\pm\sqrt{\frac{3}{7} - \frac{2}{7}\sqrt{\frac{6}{5}}}$	$\frac{18+\sqrt{30}}{36}$
	$\pm\sqrt{\frac{3}{7} + \frac{2}{7}\sqrt{\frac{6}{5}}}$	$\frac{18-\sqrt{30}}{36}$
5	0	$\frac{128}{225}$
	$\pm\frac{1}{3}\sqrt{5 - 2\sqrt{\frac{10}{7}}}$	$\frac{322+13\sqrt{70}}{900}$
	$\pm\frac{1}{3}\sqrt{5 + 2\sqrt{\frac{10}{7}}}$	$\frac{322-13\sqrt{70}}{900}$



Numerical integration

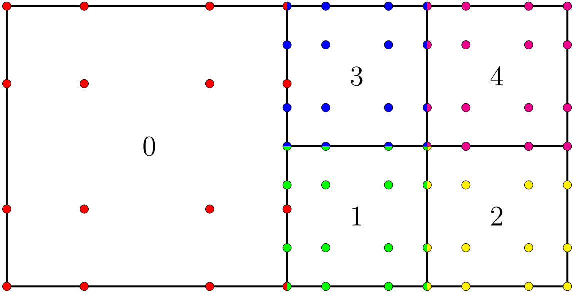
Gauss points in 2D



Tensor-product Gauss points. Weights are products of 1D-weights.

Numerical integration

- Accuracy in Gauss-(Chebyshev, Laguerre, Hermite, Legendre,...) can only be improved by increasing number of points
- Of particular interest are quadrature points for infinite intervals (Laguerre, Hermite)
- Interval partitioning superior, but only possible for $\omega \equiv 1$ (Gauss-Legendre or Gauss-Lobatto)



2D-Gauss-Lobatto integration points (also used as interpolation points).

Integration on $[0, \infty)$ (Laguerre)

How to approximate

$$\int_0^\infty g(t)\,dt,$$

where g decays rapidly enough such that the integral is finite?
Laguerre integration assumes integration weighted with e^{-t}...

should choose roots of Laguerre polynomials, which integrate upto degree $2n+1$ exactly $\int e^{-t} g(t)\,dt$ for $g \in \mathbb{P}_{2n+1}$

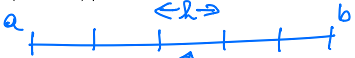
$$\int_0^\infty g(t) = \int_0^\infty e^{-t} \underbrace{\left(e^{t} g(t)\right)}_{\hat{g}(t)} dt \approx \sum_{i=0}^{n} \lambda_{in}\, e^{t_i} g(t_i)$$

roots of Laguerre polynom.

Numerical integration

Interval partitioning

Split interval $[a, b]$ into subintervals of size $h = (b-a)/n$. Basic trapezoidal sum:



$$T(h) = T^n = h\left(\frac{1}{2}(f(a) + f(b)) + \sum_{i=1}^{n-1} f(a+ih)\right)$$

trapezoidal rule on each subinterval

We can now think of what happens as $h \to 0$ (i.e., more subintervals).

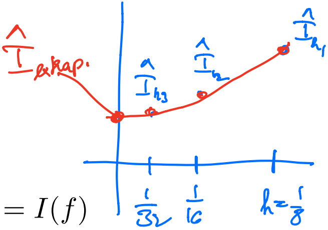
Theorem: For $f \in C^{2m+1}$ holds:

$$T(h) - \int_a^b f(t)\,dt = \tau_2 h^2 + \tau_4 h^4 + \ldots + R_{2m+2}(h) h^{2m+2},$$

which coefficients τ_i that depend on the derivatives of f at a and b, and on the Bernoulli numbers B_{2k} and $R_{2m+2}(h)$ is a remainder term that involves $f^{(2m)}$.

Numerical integration

Interval partitioning



Extrapolation:

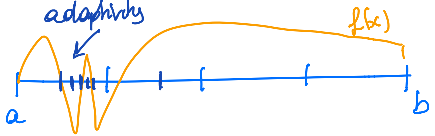
$$\lim_{h \to 0} T(h) := \lim_{n \to \infty} T^n = I(f)$$

Extrapolation uses $h_1, \ldots, h_n$ to estimate the limit.

- One can estimate the order gained by extrapolation theoretically (Thm. 9.22 in Deuflhard/Hohmann).
- Existence of an asymptotic expansion with a certain order h^p $(p \geq 1)$ can be used to improve the extrapolation.
- Extrapolation ideas with Trapezoidal rule leads to Romberg quadrature.

Numerical integration

Adaptive interval partitioning



Idea: On each sub-interval, estimate the quadrature error by either:

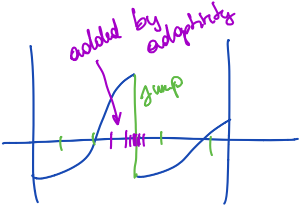
- Using a higher-order quadrature (e.g., Simpson rule), or
- Comparing the error on a subinterval with the error on a refinement

Then, subdivide the interval depending on the error estimation, and repeat. Main challenge: Derive an error estimator $\bar{\epsilon}$ that estimates the true error ϵ in the following way:

$$c_1\epsilon \leq \bar{\epsilon} \leq c_2\epsilon$$

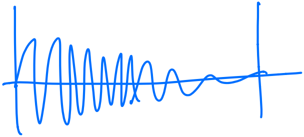
with $c_1 \leq 1 \leq c_2$. This method uses an a posteriori estimate of the error.

Numerical integration



Difficult cases for quadrature:

- (Unknown) discontinuities in f: adaptive quadrature continues to refine, which can be used to localize discontinuities



- Highly oscillating integrals
- (Weakly) singular integrals (as required, e.g., in integral methods (the fast multipole method)