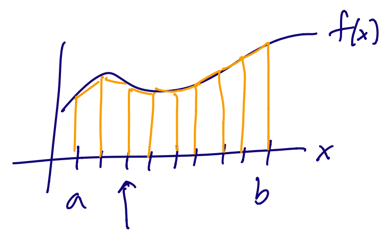
April 23, 2020

Numerical Analysis

(Composite) Trapezoidal rule for approximating the integral



$$\int_a^b f(x)\,dx \approx T_n f$$

$$= \sum_{j=1}^{n} h \cdot \frac{1}{2} \cdot \left(f(x_{j-1}) + f(x_j) \right)$$

$$= \sum_{j=0}^{n} h \cdot f(x_j) - \frac{1}{2} h \left(f(a) + f(b) \right).$$

$x_j = a + hj$

$h = \frac{b-a}{n}$

If f is periodic, then $f(a) = f(b)$, and the composite trapezoidal rule reduces to:

$$\int_a^b f(x)\,dx \approx \sum_{j=0}^{n} h \cdot f(x_j) - \underbrace{\frac{1}{2} h \left(f(x_n) + f(x_n) \right)}_{= h \cdot f(x_n)}$$

$$= \sum_{j=0}^{n-1} h \cdot f(x_j).$$

Ex: Use the trapezoidal rule to compute

$$J_0(x) = \frac{1}{\pi} \int_0^{\pi} e^{ix\cos\theta}\, d\theta$$

$$= \frac{1}{2\pi} \int_0^{2\pi} e^{ix\cos\theta}\, d\theta$$

$$\approx \frac{1}{2\pi} \frac{2\pi}{n} \sum_{j=0}^{n-1} e^{ix\cos\theta_j} \qquad \theta_j = \frac{2\pi}{n} \cdot j$$

$$= \frac{1}{n} \sum_{j=0}^{n-1} e^{ix\cos \underbrace{\frac{2\pi j}{n}}}$$

quadrature node

Converges extremely fast to exact solution. (Euler-MacLaurin Thm that predicts this.)

Gaussian Quadrature Rules

General form of quadrature rules: $\displaystyle\int_a^b f(x)\,dx \approx \sum_{j=1}^{n} w_j f(x_j)$.

n weights n nodes.

This rule can be made exact for certain functions.

Ex: If $\displaystyle\int_a^b \cos x \, dx = w_j \cos x_j$ then x_j can be anything,

and $\displaystyle w_j = \frac{1}{\cos x_j} \int_a^b \cos x \, dx$

$= $ a number.

Once the nodes x_j are determined, finding the correct w_j's such that $f_1, .., f_n$ are integrated exactly requires only solving a linear system.

Ex: If the quadrature rule $\displaystyle\int_a^b f(x)\,dx = \sum_{j=1}^{n} w_j f(x_j)$

is exact for $f = f_1, f_2, \dots f_n$ then it must be the case that

$$\sum_{j=1}^{n} w_j f_1(x_j) = \int_a^b f_1(x)\,dx = I_1$$

$$\sum_{j=1}^{n} w_j f_2(x_j) = \int_a^b f_2(x)\,dx = I_2$$

$$\vdots$$

$$\sum_{j=1}^{n} w_j f_n(x_j) = I_n$$

If the nodes are specified, then the rule can be made exact for n functions by solving a linear system.

$\Rightarrow$ This can be written as:

$$\begin{pmatrix} f_1(x_1) & f_1(x_2) & \cdots & f_1(x_n) \\ \vdots & & & \\ f_n(x_1) & & \cdots & f_n(x_n) \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{pmatrix} = \begin{pmatrix} I_1 \\ I_2 \\ \vdots \\ I_n \end{pmatrix}.$$

A quadrature rule of the form $\int_a^b f(x)\,dx \approx \sum_{j=1}^{n} w_j f(x_j)$ is called <u>GAUSSIAN</u> if it is exact for $2n$ linearly independent functions.

How is this possible? Nonlinear optimization/root finding

I.e. find weights <u>and</u> nodes simultaneously so that $2n$ integrals are computed exactly.

<u>Special Case</u> Integrating the functions $\underbrace{1, x, x^2, \ldots, x^{2n-1}}$ exactly.

polynomials of degree less than or equal to $2n-1$.

The key to deriving Gaussian quadrature rules for polynomials is <u>orthogonal polynomials</u>.

(*) <u>Theorem</u> If $x_1, \ldots, x_n$ are the zeros (roots) of P_n, the degree n Legendre polynomial, then the formula:

$$\int_a^b f(x)\,dx \approx \sum_{j=1}^{n} w_j f(x_j)$$

when $w_j = \int_a^b q_j(x)\,dx$, $q_j(x) = \prod_{\ell \neq j} \frac{x - x_\ell}{x_j - x_\ell}$

is <u>exact</u> for polynomials of degree $2n-1$ or less.

$\Rightarrow$ <u>Exact</u> for $2n$ linearly independent functions!

Legendre Polynomials are the set of polynomials which are orthogonal on the interval $[-1,1]$, with $P_0(x) = 1$, $P_1(x) = x$, and $P_{n+1}(x) = \frac{2n+1}{n+1} x P_n(x) - \frac{n}{n+1} P_{n-1}(x)$. } Three term recurrence.

This three term recurrence can be derived from Gram-Schmidt:

$$P_0(x) = 1 \quad , \quad \hat{P}_0(x) = \frac{1}{\|P_0\|_2} = \frac{1}{\sqrt{\int_{-1}^{1} 1\,dx}} = \frac{1}{\sqrt{2}}$$

$$P_1(x) = x \quad , \quad \hat{P}_1(x) = \frac{x}{\|P_1\|_2} = \frac{x}{\sqrt{\int_{-1}^{1} x^2\,dx}} = \sqrt{\frac{3}{2}}\, x .$$

Gram-Schmit say set (for example) $q_2(x) = x^2$, (a new linearly independent function and project off the P_0, P_1 components)

So $P_2(x) = x^2 - (q_2, \hat{P}_1)\hat{P}_1 - (q_2, \hat{P}_0)\hat{P}_0$

and in general, $P_n = x^n - \sum_{j=0}^{n-1} (q_n, \hat{P}_j)\hat{P}_j$

$q_n(x) = x^n$

Instead of using $q_n = x^n$, set $\underbrace{q_n(x)}_{\text{degree } n \text{ polynomial.}} = x \cdot \underbrace{\hat{P}_{n-1}(x)}_{\text{degree } n-1 \text{ polynomial}}$.

So for $n = 2, 3, 4 \ldots$

Set (1) $P_n(x) = x\hat{P}_{n-1}(x) - \sum_{j=0}^{n-1} \underbrace{(x\hat{P}_{n-1}, \hat{P}_j)}_{= \int_{-1}^{1} x\, \hat{P}_{n-1}(x) \cdot \hat{P}_j(x)\, dx} \hat{P}_j$

(2) and then normalize: $\hat{P}_n = \frac{P_n}{\|P_n\|_2}$.

Why is this procedure better?

This scheme is better because <u>by construction</u> P_{n-1} is orthogonal to all polynomials of degree $\leq n-2$.

Any polynomial of degree $\leq n-2$ can be written as a linear combination of $P_0, P_1, .., P_{n-2}$. These functions are all orthogonal to P_{n-1}.

So $\Rightarrow$ $\left(x\hat{P}_{n-1}, \hat{P}_j\right) = \left(\hat{P}_{n-1}, x\hat{P}_j\right) = 0$ if $j \leq n-3$.

This means that the Gram-Schmidt process simplifies:

$$P_n(x) = x\hat{P}_{n-1}(x) - \left(x\hat{P}_{n-1}, \hat{P}_{n-1}\right)\hat{P}_{n-1} - \left(x\hat{P}_{n-1}, \hat{P}_{n-2}\right)\hat{P}_{n-2}.$$

$$\underbrace{= \left(x - \left(x\hat{P}_{n-1}, \hat{P}_{n-1}\right)\right)\hat{P}_{n-1} - \left(x\hat{P}_{n-1}, \hat{P}_{n-2}\right)\hat{P}_{n-2}}$$

Three term recurrence, meaning P_n can be calculated from P_{n-1} and P_{n-2}.

As we had before: $P_{n+1}(x) = \frac{2n+1}{n+1}x P_n(x) - \frac{n}{n+1}P_{n-1}(x)$.

Back to Gaussian Quadratures:

<u>Proof</u> [of Theorem (*)] Let f be a polynomial of degree $\leq 2n-1$.

This implies that $f = q P_n + r$ where

$\deg(q) \leq n-1$ $\deg(P_n) = n$ $\deg(r) \leq n-1$ $\Big\}$ by polynomial long division

So, if x_j is a root of P_n, then
$$f(x_j) = q(x_j)\underbrace{\cancel{P(x_j)}}_{=0} + r(x_j) = r(x_j).$$

Now integrate:

$$\int_{-1}^{1} f(x)\,dx = \overbrace{\int_{-1}^{1} q(x) P_n(x)\,dx}^{=0 \text{ since } \deg(q) \le n-1} + \int_{-1}^{1} r(x)\,dx$$

$$= \int_{-1}^{1} r(x)\,dx.$$

Now: given the choice of our weights earlier,

$$\int_{-1}^{1} r(x)\,dx = \sum_{j=1}^{n} w_j\, r(x_j) = \sum_{j=1}^{n} w_j\, f(x_j).$$

But w_j was chosen to give an exact integral for any polynomial of degree $\le n-1$. (From the Newton-Cotes construction.)

<u>To summarize</u> On the interval $[-1,1]$, the n-point Gaussian Quadrature integrates $1, x, x^2, \dots, x^{2n-1}$ exactly, where the nodes x_j are the roots of P_n and the weights can be precomputed from Newton-Cotes.

<u>Another Example</u>: Chebyshev Polynomials

Chebyshev polynomials are orthogonal $[-1,1]$ with weight function $w(x) = \dfrac{1}{\sqrt{1-x^2}}$.

Let x_j be the roots of T_n. Then the weights are:

$$w_j = \int_{-1}^{1} q_j(x)\, w(x)\,dx, \qquad q_j(x) = \prod_{l \ne j} \frac{x - x_l}{x_j - x_l}.$$

Then the quadrature rule:

$$\Rightarrow \int_{-1}^{1} f(x) \frac{1}{\sqrt{1-x^2}}\,dx \approx \sum_{j=1}^{n} w_j\, f(x_j)$$

is exact for f a polynomial of degree $\le 2n-1$.