UCSB, Physics 129L, Computational Physics
Lecture notes, Week 8

Zihang Wang (UCSB), zihangwang@ucsb.edu

March 3, 2025

Contents

1 Differential Equation

Numerical methods used in physics are fundamentally connected to solving $n-th$ order ordinary/partial differential equations,

$$D_{\mathbf{x}}^n F(\mathbf{x}) = G(\mathbf{x}), \tag{1}$$

where $D_{\mathbf{x}}$ represents **multi-variable differential operator**, and $G(\mathbf{x}), F(\mathbf{x})$ are the **driven function** and the **response function**, respectively.

1.1 Poisson equation and Greens' function

The **Poisson equation** in electrostatics governs the electric potential $V(\mathbf{x})$ due to a charge distribution $\rho(\mathbf{x})$:

$$\partial_{\mathbf{x}}^2 V(\mathbf{x}) = -\frac{\rho(\mathbf{x})}{\epsilon}, \tag{2}$$

where ϵ is the **permittivity** of an environment. This equation relates the electric potential to the charge density distribution. This is second order equation,

A common method for solving the Poisson equation is the Green's function approach. The Green's function $G(\mathbf{x}, \mathbf{x}')$ satisfies:

$$\partial_{\mathbf{x}}^2 G(\mathbf{x}, \mathbf{x}') = -\delta(\mathbf{x} - \mathbf{x}'), \tag{3}$$

where $\delta(\mathbf{x} - \mathbf{x}')$ is the Dirac delta function. You might see an extra minus sign that is different from in quantum mechanics. It is convention. Using Green's function, the solution to Poisson's equation is given by:

$$V(\mathbf{x}) = \frac{1}{\epsilon} \int_{\Omega} G(\mathbf{x}, \mathbf{x}') \rho(\mathbf{x}') \, d^3x' + \int_{\partial\Omega} \left(G(\mathbf{x}, \mathbf{x}') \frac{\partial V(\mathbf{x}')}{\partial n} - V(\mathbf{x}') \frac{\partial G(\mathbf{x}, \mathbf{x}')}{\partial n} \right) dS(\mathbf{x}'), \tag{4}$$

where first integral represents the **superposition principle**, where the potential at $\mathbf{x}$ is obtained by summing contributions from all charge elements. The second integral $\partial\Omega$ accounts for the boundary conditions. It involves two parts: $G(\mathbf{x}, \mathbf{x}') \frac{\partial V(\mathbf{x}')}{\partial n}$ represents the influence of the **normal derivative** of the potential on the boundary. $V(\mathbf{x}') \frac{\partial G(\mathbf{x}, \mathbf{x}')}{\partial n}$ accounts for the effect of the potential on the normal derivative of the Green's function at the boundary.

- **Dirichlet Boundary Condition:** The Dirichlet boundary condition specifies the value of the potential $V(\mathbf{x})$ on the boundary $\partial\Omega$. Physically, this corresponds to a situation where the potential is fixed (e.g., a grounded conductor where $V = 0$ on the boundary),

$$V(\mathbf{x}) = 0 \quad \text{and } G(\mathbf{x}) = 0 \quad \text{on} \quad \partial\Omega.$$

- **Neumann Boundary Condition:** The Neumann boundary condition specifies the normal derivative of the potential $\frac{\partial V(\mathbf{x})}{\partial n}$ on the boundary. Physically, this corresponds to a situation where the **normal component of the electric field** is fixed, which is related to the flux of charge through the boundary. For example, in an insulated boundary, no charge flows across the boundary,

$$\frac{\partial V(\mathbf{x})}{\partial n} = 0 \quad \text{and } \frac{\partial G(\mathbf{x})}{\partial n} = 0 \quad \text{on} \quad \partial\Omega.$$

- **Mixed Boundary Condition:** A mixed boundary condition involves both a Dirichlet and a Neumann condition, specifying both the value of the potential and its normal derivative at different parts of the boundary. This may occur in cases where some parts of the boundary are held at a constant potential, while other parts have a prescribed normal derivative (e.g., where the electric field is fixed),

$$V(\mathbf{x}) = V_0 \quad \text{on} \quad \partial\Omega_1, \quad \frac{\partial V(\mathbf{x})}{\partial n} = 0 \quad \text{on} \quad \partial\Omega_2.$$

For example, i**n free space**, the Green's function in three dimensions is:

$$G(\mathbf{x}, \mathbf{x}') = \frac{1}{4\pi|\mathbf{x} - \mathbf{x}'|}. \tag{5}$$

Thus, the potential can be expressed as:

$$V(\mathbf{x}) = \frac{1}{4\pi\epsilon} \int \frac{\rho(\mathbf{x}')}{|\mathbf{x} - \mathbf{x}'|} d^3x'. \tag{6}$$

This is the usual formula in finding electrostatic potential due to a charge distribution.

1.2 Laplace's equation and harmonic functions

When the charge density is zero, i.e., $\rho(\mathbf{x}) = 0$, Poisson's equation reduces to **Laplace's equation**,

$$\nabla^2 V = 0. \tag{7}$$

Laplace's equation describes the behavior of electric potential in charge-free regions. Solutions to this equation are **harmonic functions**.

The value of a harmonic function at a point is the **average of its values over any surrounding sphere**,

$$V(\mathbf{x}) = \frac{1}{|\partial B_r(\mathbf{x})|} \oint_{\partial B_r(\mathbf{x})} V(\mathbf{x}') dS, \tag{8}$$

where $B_r(\mathbf{x})$ is a sphere of radius r centered at $\mathbf{x}$, and dS is the surface element.

A non-constant **harmonic function does not attain a local maximum or minimum inside a domain and extrema occur on the boundary**,

$$\max_{\mathbf{x} \in \Omega} V(\mathbf{x}) = \max_{\mathbf{x} \in \partial\Omega} V(\mathbf{x}). \tag{9}$$

If a solution to Laplace's equation satisfies **given boundary conditions**, it is **unique**. That is, if two functions V_1 and V_2 satisfy Laplace's equation with the same boundary conditions,

$$V_1(\mathbf{x}) = V_2(\mathbf{x}) \quad \text{for all } \mathbf{x} \in \Omega. \tag{10}$$

1.3 Special Variation and Gradient

The simplest method to approximate differentiation is to use the concept of **finite difference**. Let's define the infinitesimal step $\mathbf{h} = \mathbf{x} - \mathbf{x_A}$, and the **remainder** (difference between the Taylor expansion and the exact) has the same order as $\mathbf{h}$. Let's consider a general Taylor expansion upto **second-order accurate**,

$$F(\mathbf{x_A} \pm \mathbf{h}) - F(\mathbf{x_A}) \approx \pm \mathbf{h} \cdot \partial_{\mathbf{x}} F(\mathbf{x})|_{\mathbf{x_0}} + \frac{1}{2} \mathbf{h} \cdot (\mathbf{H}F(\mathbf{x})|_{\mathbf{x_0}}) \mathbf{h} + \mathcal{O}\left(|\mathbf{h}|^3\right), \tag{11}$$

where $\mathbf{H}F(\mathbf{x})|_{\mathbf{x_0}}$ is the **Hessian** at $\mathbf{x}_0$. Similarly, the **Laplacian** (scalar) can be calculated via the **trace of the Hessian**.

The **forward difference** is defined as the following,

$$F(\mathbf{x_A} + \mathbf{h}) - F(\mathbf{x_A}) \approx \mathbf{h} \cdot \partial_{\mathbf{x}} F(\mathbf{x})|_{\mathbf{x_0}} + \mathcal{O}\left(|\mathbf{h}|^2\right). \tag{12}$$

Taking another forward difference to approximate the Hessian,

$$\mathbf{h}\cdot[\partial_{\mathbf{x}}F(\mathbf{x_A}+\mathbf{h})-\partial_{\mathbf{x}}F(\mathbf{x_A})]\approx F(\mathbf{x_A}+2\mathbf{h})-2F(\mathbf{x_A}+\mathbf{h})+F(\mathbf{x_A})+\mathcal{O}(|\mathbf{h}|^3).$$

Thus, the Hessian via forward differences is:

$$\mathbf{h}\cdot(\mathbf{H}F(\mathbf{x})|_{\mathbf{x_0}})\mathbf{h}\approx F(\mathbf{x_A}+2\mathbf{h})-2F(\mathbf{x_A}+\mathbf{h})+F(\mathbf{x_A})+\mathcal{O}(|\mathbf{h}|^3).$$

The factor of two in the Hessian is canceled by the factor 2 in both $F(\mathbf{x_A}+2\mathbf{h})$ and $2F(\mathbf{x_A}+\mathbf{h})$. The approximation is **first-order accurate** ($\mathcal{O}(|\mathbf{h}|)$).

The **backward difference** has the following form,

$$F(\mathbf{x_A})-F(\mathbf{x_A}-\mathbf{h})\approx\mathbf{h}\cdot\partial_{\mathbf{x}}F(\mathbf{x})|_{\mathbf{x_0}}+\mathcal{O}\left(|\mathbf{h}|^2\right). \tag{13}$$

The Hessian via backward differences is:

$$\mathbf{h}\cdot(\mathbf{H}F(\mathbf{x})|_{\mathbf{x_0}})\mathbf{h}\approx F(\mathbf{x_A}-2\mathbf{h})-2F(\mathbf{x_A}-\mathbf{h})+F(\mathbf{x_A})+\mathcal{O}(|\mathbf{h}|^3).$$

The **mid-point (central) difference** has the following form, and

$$F(\mathbf{x_A}+\frac{\mathbf{h}}{\mathbf{2}})-F(\mathbf{x_A}-\frac{\mathbf{h}}{\mathbf{2}})\approx\mathbf{h}\cdot\partial_{\mathbf{x}}F(\mathbf{x})|_{\mathbf{x_0}}+\mathcal{O}\left(|\mathbf{h}|^3\right). \tag{14}$$

We note that it has an order $|\mathbf{h}|^3$, and it is because the second order term is symmetric and is subtracted from the above mid-point difference.

The Hessian via the central differences,

$$\mathbf{h}\cdot(\mathbf{H}F(\mathbf{x})|_{\mathbf{x_0}})\mathbf{h}\approx F(\mathbf{x_A}-\mathbf{h})+F(\mathbf{x_A}+\mathbf{h})-F(\mathbf{x_A})+\mathcal{O}(|\mathbf{h}|^4).$$

We note that it has an order $|\mathbf{h}|^3$. The above mid-point approximation provides an important insight: **Certain combinations of different n-th accurate approximation leads $n+1$-th accurate approximation. This is an essential idea in various numerical integration techniques, and we will discuss it in the following.**

Numerical Integration (Quadrature)

Let's first consider a single-variable integral,

$$I=\int_{x_A}^{x_B}F(x)dx, \tag{15}$$

that we want to approximate. Suppose we split this integral into N equal subintervals with $h_N=(x_B-x_A)/(N)$. We approximate the integral as the following deterministic sum of functions evaluated at different $F(x_j)$,

$$I=\sum_{j=1}^{N}h_NF(x_j)+r_N, \tag{16}$$

where we define the **remainder** of the above integral as r_N.

For example, consider the Taylor expansion of a function $f(x)$ around the midpoint x_m:

$$f(x) = f(x_m) + f'(x_m)(x - x_m) + \frac{f''(x_m)}{2}(x - x_m)^2 + \frac{f^{(3)}(x_m)}{6}(x - x_m)^3 + \cdots$$

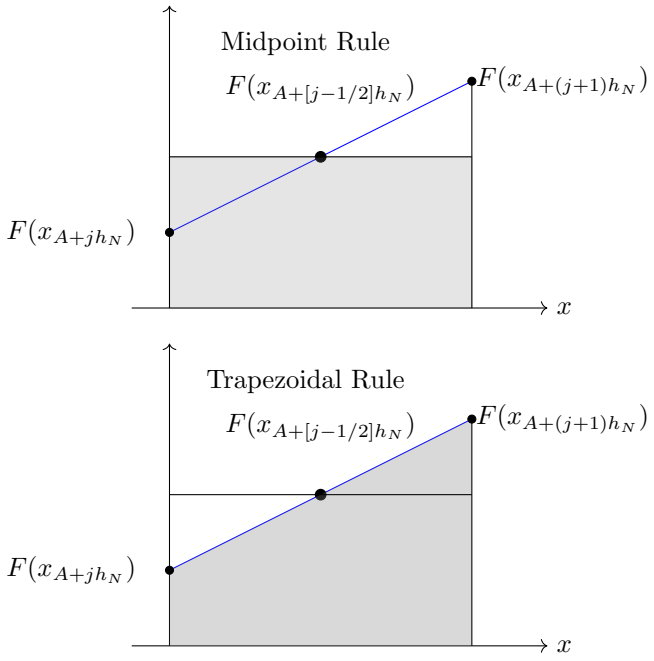
When you integrate $f(x)$ over a symmetric interval around x_m, **the terms with $(x - x_m)$ and $(x - x_m)^3$ (and all odd powers) vanish because the positive and negative parts cancel each other out**. What remains are the even-powered terms, which directly lead to error terms proportional to h^2 for the trapezoidal rule and h^4 for Simpson's rule.

The **midpoint rule** approximates the integral by the **area of a rectangle** whose base has length h_N, whose **height is the value of $F(x)$ at the midpoint**,

$$I = M_N + r_N = \sum_{j=1}^{N} M_{N,j} + \mathcal{O}\left(1/N^2\right), \tag{17}$$

upto the quadratic remainder $r_N \sim \mathcal{O}\left(1/N^2\right)$, and

$$M_{N,j} = h_N F(x_{A+[j-\frac{1}{2}]h_N}). \tag{18}$$



Instead of using the area of a rectangle, we can use **the area of trapezoids**. The **single midpoint is replaced by the average of two end points**,

$$F\left(x_{A+\left(j-\frac{1}{2}\right)h_N}\right) \approx \frac{1}{2}\left[F(x_{A+(j-1)h_N}) + F(x_{A+jh_N})\right],$$

we can substitute this into the midpoint sum:

$$T_{N,j} = h_N \frac{1}{2} \left[F(x_{A+(j-1)h_N}) + F(x_{A+jh_N}) \right].$$

We observe that this sum includes each function value twice, except for the first and last points:

$$I = T_N + r_N = \sum_{j=1}^{N} T_{N,j} + r_N = \frac{h_N}{2} \left[F(x_A) + 2 \sum_{j=1}^{N-1} F(x_{A+jh_N}) + F(x_{A+Nh_N}) + \mathcal{O}\left(1/N^2\right) \right].$$

This is the **composite trapezoidal rule**.

We see both approximations have an order $1/N^2$ remainder, therefore are both **exact for constant and linear functions: When $F(x)$ is a linear polynomial, the above methods give exact results**. In other words, the Taylor expansion is exact upto the first order,

$$I = I^{(0)} + I^{(1)} x + r_N = \begin{cases} M_N \\ T_N \end{cases} \quad + r_N. \tag{19}$$

What about functions beyond the linear order, i.e.,

$$I = I^{(0)} + I^{(1)} x + 3I^{(2)} x^2 + 4I^{(3)} x^3 + r_N. \tag{20}$$

Let's evaluate the integral exactly over a fixed interval (scaled to 1 for connivance as the boundary does not change the scaling),

$$I^{(2)} = \int_0^1 x^2 dx = \frac{1}{3}, \quad I^{(3)} = \int_0^1 x^3 dx = \frac{1}{4}, \tag{21}$$

which is exact, and the M and T approximations are given as the following,

$$M^{(2)} = 1 \frac{1}{2^2} = \frac{1}{4}, \quad r = I^{(2)} - M^{(2)} = \frac{1}{12}, \quad T^{(2)} = 1 \frac{0^2 + 1^2}{2} = \frac{1}{2}, \quad r = I^{(2)} - T^{(2)} = -\frac{1}{6}, \tag{22}$$

and,

$$M^{(3)} = 1 \frac{1}{2^3} = \frac{1}{8}, \quad r = I^{(3)} - M^{(3)} = \frac{1}{8}, \quad T^{(3)} = 1 \frac{0^3 + 1^3}{2} = \frac{1}{2}, \quad r = I^{(3)} - T^{(3)} = -\frac{2}{8}. \tag{23}$$

In the above two examples, it seems like the **midpoint rule** and the **trapezoidal Rule** can work together to give an order of $1/N^4$ remainder, i.e., we solve the equation and generalize into different index,

$$I_{N,j} - T_{N,j} = -2(I_{N,j} - M_{N,j}) \rightarrow I_{N,j} = \frac{2}{3} M_{N,j} + \frac{1}{3} T_{N,j}, \tag{24}$$

where the $I_{N,j}$ is the exact integral value around j.

$$\begin{aligned} I_{N,j} &= \frac{h_N}{3} \left(2h_N F(x_{A+[j-\frac{1}{2}]h_N}) + \frac{1}{2} \left[F(x_{A+(j-1)h_N}) + F(x_{A+jh_N}) \right] \right) + r_N \\ &= \frac{h_N}{6} \left(4h_N F(x_{A+[j-\frac{1}{2}]h_N}) + \left[F(x_{A+(j-1)h_N}) + F(x_{A+jh_N}) \right]. \right) + \mathcal{O}(1/N^4), \end{aligned} \tag{25}$$

which is the famous **Simpson's rule**. Classically, Simpson's rule is derived by using a parabola to interpolate the given function at the two endpoints, x_A and x_B, and the midpoint, $(x_B + x_A)/2$ and integrating the parabola exactly (therefore, a third power after integration). Therefore, we got an order of $1/N^4$ remainder.

The above result can be used to define the **composite Simpson's rule** where we subdivide into $N/2$ mid points, and preform a sum of all subinterval Simpson's rules,

$$\begin{aligned} I &= \sum_{j=1}^{N} \left(\frac{2}{3} M_{N,j} + \frac{1}{3} T_{N,j} \right) \\ &= \frac{h_N}{6} \left\{ F(x_A) + F(x_B) + 4 \sum_{j=1}^{N} F\left(x_{A+(j-\frac{1}{2})h_N} \right) + 2 \sum_{j=1}^{N-1} F\left(x_{A+jh_N} \right) \right\} + \mathcal{O}\left(\frac{1}{N^4} \right) \end{aligned} \tag{26}$$

where we get a $r_N \sim \mathcal{O}(1/N^4)$ advantage since $h_N = (x_B - x_A)/N$.

Richardson extrapolation and Romberg integration

Richardson extrapolation involves taking two or more approximations of a quantity using different step sizes, and then combine these approximations to obtain a more accurate estimate. Following the same idea, we can cancel out the higher order remainder by double the number of grid points by using the Richardson extrapolation: $N \to 2N \to 4N \ldots$, which can be used in conjunction with the composite Simpson's rule.

Let's first define the integral value by the order of the accuracy,

$$R_{i,1} = I_{(2^{i-1})N}, \tag{27}$$

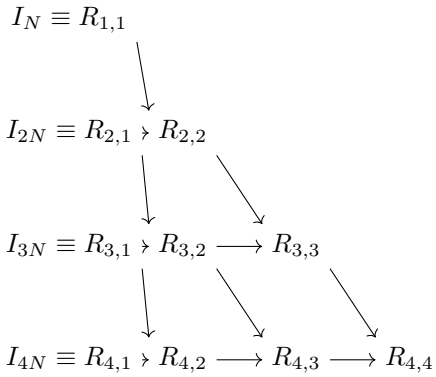
where in this case, i labels the **grid size expansion in the power of 2**. The integral term $I_{(2^{i-1})N}$ represents any numerical methods,

For example, $R_{3,3}$ means we expand the grid size from $N \to 4N$, with a 6-th order of accuracy. Similar to the Simpson's rule, we have,

$$2^j (R_{i,j} - R_{i,j-1}) = (R_{i,j} - R_{i-1,j-1}) \to R_{i,j} = \frac{1}{2^{2j} - 1} \left(2^{2j} R_{i,j-1} - R_{i-1,j-1} \right).$$

where j labels the j-th order accuracy in the **multiple** of 2. You should note that there is a factor of $2^2 = 4$: The remainder only contains even order, upon the subdivision, the scaling becomes $h^2 \to (h/2)^2$. You may find the difference between the above expression and the Simpson's rule that uses both mid-point rule and trapezoid rule. Both M and T in the above use the same subinterval while in Richardson extrapolation, we relate expressions between different subintervals.

This recurrence process is called the **Romberg integration**: We generates the **composite trapezoidal rule** by increasing $N \to 2N$, then subtract a lower order one by jumping between even intervals, as showing in the figure below.



1.4 Example on Romberg Integration

We aim to compute the value of the **error function**,

$$I = \int_0^1 e^{-x^2} dx,$$

an integral that cannot be solved analytically using elementary functions.

Subinterval, 2^0: With $n = 1$ subinterval ($h = 1/1 = 1$), we use the trapezoidal rule:

$$R_{1,1} = \frac{h}{2}[f(0) + f(1)] = 0.68393972058572,$$

with an absolute error of 6.3×10^{-2}.

Subinterval, 2^1: Bisecting into two subintervals ($n = 2, h = 0.5$), we apply the composite trapezoidal rule:

$$R_{2,1} = \frac{h}{2}[f(0) + 2f(0.5) + f(1)] = 0.73137025182856,$$

with an absolute error of 1.5×10^{-2}.

Using Richardson extrapolation:

$$R_{2,2} = R_{2,1} + \frac{R_{2,1} - R_{1,1}}{3} = 0.74718042890951,$$

with an absolute error of 3.6×10^{-4}.

Subinterval, 2^2: Further bisecting ($n = 4, h = 0.25$), we compute the following,

$$R_{3,1} = \frac{h}{2}[f(0) + 2(f(0.25) + f(0.5) + f(0.75)) + f(1)] = 0.74298409780038,$$

with an absolute error of 3.8×10^{-3}.

Applying Richardson extrapolation again:

$$R_{3,2} = R_{3,1} + \frac{R_{3,1} - R_{2,1}}{3} = 0.74685537979099,$$

with an absolute error of 3.1×10^{-5}.

Subinterval, 2^3:

$$R_{3,3} = R_{3,2} + \frac{R_{3,2} - R_{2,2}}{15} = 0.74683370984975,$$

with an absolute error of 9.6×10^{-6}.

Romberg integration achieves high accuracy by systematically and iteratively reducing errors through extrapolation.

1.5 Polynomial Interpolation

Let's consider a general case where we approximate a function $F(x)$ over an interval $x \in [x_A, x_B]$ by a n-th degree polynomial $p_n(x)$ that exactly matches $F(x)$ at n distinct nodes $x_1, x_2, \ldots, x_n$. In other words, we require the following,

$$p_n(x_i) = F(x_i) \quad \text{for } i = 1, \ldots, n.$$

A common way to write this interpolating polynomial is using **Lagrange interpolation**,

$$p(x) = \sum_{i=1}^{n} F(x_i)\, \ell_i(x),$$

where the **Lagrange basis polynomials** $\ell_i(x)$ are defined as,

$$\ell_i(x) = \prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j}.$$

Each $\ell_i(x)$ is constructed so that it equals 1 at $x = x_i$ and 0 at all other nodes x_j (with $j \neq i$).

We define the difference between the two by the **error polynomial**,

$$C \cdot r(x) = F(x) - p_n(x) = C \cdot \prod_{i=1}^{n} (x - x_i), \tag{28}$$

where C is a multiplicative constant.

1.6 Gaussian-type Quadrature

Gaussian quadrature is a numerical integration method that approximates an integral as **a weighted sum of function values at specially chosen nodes**.

A definite integral of $F(x)$ can be projected to a new basis set $g(x)$ with a given weight function $w(x)$,

$$I = \int_a^b F(x)dx = \int_a^b w(x)g(x)dx,$$

where we define the new basis set in a similar way as we did in **Monte Carlo integration**,

$$g(x) = \frac{f(x)}{w(x)}. \tag{29}$$

We approximate $g(x)$ using **Lagrange interpolation** via a n-th order polynomial. We select n quadrature nodes x_i,

$$g(x) \approx \sum_{i=1}^{n} g(x_i)\ell_i(x), \tag{30}$$

where $\ell_i(x)$ are the **Lagrange basis polynomials** discussed previously. We then substitute the above approximation into the integral:

$$I = \int_{x_a}^{x_b} w(x)g(x)dx \approx \int_{x_a}^{x_b} w(x)\sum_{i=1}^{n} g(x_i)\ell_i(x)dx = \sum_{i=1}^{n} g(x_i)W_i\frac{w(x_i)}{w(x_i)} = \sum_{i=1}^{n} W_i f(x_i), \tag{31}$$

where we define the **quadrature weight** associated with node x_i,

$$W_i = \int_{x_a}^{x_b} w(x)\ell_i(x)dx. \tag{32}$$

These are then approximated by a sum of function values at specified points x_i, multiplied by some weights w_i. While the above approximation reduces the integral into a sum of nodes, we still need to find the coordinates of those nodes such that the resulting sum approximates an integral optimally via an n-th order polynomial,

$$\int_a^b w(x)g(x)dx - \sum_{i=1}^{n} g(x_i)W_i = 0, \tag{33}$$

the above expression is a system of equations that exactly approximates $2n-1$-th order polynomials (it is $2n-1$ because we start with the polynomial $x^0, x^1, \cdots, x^{2n-1}$.

The above $2n$ system of equations can be solved such that the nodes are given given by the roots of a n-th order orthogonal polynomial, uniquely determined by the weight $w(x)$.

1.7 Gauss-Legendre quadrature

For **Gauss-Legendre quadrature**, we can approximate

$$\int_{-1}^{1} W(x)g(x)dx \approx \sum_{i=1}^{n} w_i g(x_i),$$

where the weighting function is $w(x) = 1$. In this case, $F(x) = g(x)$.

Standard **Gaussian quadrature is set up to work on the interval** $[-1, 1]$**.** To integrate a function $f(x)$ over an arbitrary interval $[a, b]$, you perform a change of variable to map $[a, b]$ to $[-1, 1]$.

For general intervals, we can use $x = \frac{b-a}{2}t + \frac{a+b}{2}$ to map $[a, b]$ to $[-1, 1]$. The the differential is given by $dx = \frac{b-a}{2}\,dt$. This gives the following approximation,

$$I \approx \frac{b-a}{2} \sum_{i=1}^{n} w_i\, f\left(\frac{b-a}{2}t_i + \frac{a+b}{2}\right).$$

Let's consider the case where we Taylor expand the function $g(x)$ upto $2n - 1 = 4 - 1$-th order polynomial,

$$\begin{aligned}
\int_{-1}^{1} dx &= 2 = w_1 + w_2 \\
\int_{-1}^{1} \mathbf{x} dx &= 0 = w_1 x_1 + w_2 x_2 \\
\int_{-1}^{1} \mathbf{x}^2 dx &= \frac{2}{3} = w_1 \mathbf{x}_1^2 + w_2 \mathbf{x}_2^2 \\
\int_{-1}^{1} \mathbf{x}^3 dx &= 0 = w_1 \mathbf{x}_1^3 + w_2 \mathbf{x}_2^3
\end{aligned} \tag{34}$$

The above system of equations gives,

$$\begin{aligned}
x_1 = -x_2 &= \frac{1}{\sqrt{3}} \\
w_1 = w_2 &= 1
\end{aligned} \tag{35}$$

The choice of quadrature weights w_i and quadrature nodes x_i is unique and allows the rule to integrate polynomials of degree $2n - 1$ exactly.

For higher orders ($n > 2$), solving the system becomes more difficult due to

non-linearity. For example, consider the following case,

$$\begin{aligned}
\int_{-1}^{1} dx &= 2 = w_1 + w_2 + w_3 \\
\int_{-1}^{1} \mathbf{x} dx &= 0 = w_1 x_B + w_2 x_2 + w_3 x_3 \\
\int_{-1}^{1} \mathbf{x}^2 dx &= \frac{2}{3} = w_1 \mathbf{x}_1^2 + w_2 \mathbf{x}_2^2 + w_3 \mathbf{x}_3^2 \\
\int_{-1}^{1} \mathbf{x}^3 dx &= 0 = w_1 \mathbf{x}_1^3 + w_2 \mathbf{x}_2^3 + w_3 \mathbf{x}_3^3 \\
\int_{-1}^{1} \mathbf{x}^4 dx &= \frac{2}{5} = w_1 \mathbf{x}_1^4 + w_2 \mathbf{x}_2^4 + w_3 \mathbf{x}_3^4 \\
\int_{-1}^{1} \mathbf{x}^5 dx &= 0 = w_1 \mathbf{x}_1^5 + w_2 \mathbf{x}_2^5 + w_3 \mathbf{x}_3^5
\end{aligned} \tag{36}$$

The solution is turns out to be the roots of the following **Legendre polynomial**,

$$P_3(\mathbf{x}) = \frac{1}{2}(5x^3 - 3x), \;\; x_B = -x_3 = \sqrt{\frac{3}{5}}, \;\; x_2 = 0 \tag{37}$$

By plug in the roots of the **Legendre polynomial** into the system of equations, we can compute the weight,

$$w_i = \frac{2}{(1 - x_i^2)[\partial_x P_n(x_i)]^2}, \quad P_M(x) = \frac{1}{2^n n!} \frac{d^M}{dx^M} \left[(x^2 - 1)^M\right].$$

. where $\partial_x P_n(x_i)$ is the derivative of the Legendre polynomial of degree M.

1.8 Newton's method

The positions of an order M Gauss-Legendre Quadrature are calculated by finding the roots of an order-M Legendre polynomial,

$$P_M(x) = 0.$$

To numerically find those roots (M of them), we can use Newton's method:

$$x_{n+1} = x_n - \frac{P_M(x_n)}{\partial_x P_M(x_n)}.$$

You should be careful with the initial guess.