PHYS 688: Numerical Methods for AstroPhysics
Homework #1: Integration & Differentiation

Marina von Steinkirch

February 20, 2013

Q.1 **(Numerical Derivatives) Exploration of two different approaches to derive numerical derivatives:**

(a) `Derive a second-order accurate` *one-sided* `derivative at` $x = x_i$ `using the function value at the 3 points` x_i, x_{i+1}, `and` x_{i+2}, `assuming that the spacing,` Δx, `is constant.`

Consider a discrete set of data represented by three points x_i, x_{i+1}, and x_{i+2} and with a constant spacing

$$\Delta x = x_{i+2} - x_{i+1} = x_{i+1} - x_i,$$

and

$$2\Delta x = x_{i+2} - x_i.$$

Consider that the function we want to calculate the derivative (at x_i) is known at the grid points:

$$f_i = f(x_i),$$

$$f_{i+1} = f(x_{i+1}) = f(x_i + \Delta x),$$

and

$$f_{i+2} = f(x_{i+2}) = f(x_i + 2\Delta x).$$

The mathematical definition of the derivative of a function $f(x)$, where h is the step size, is

$$\frac{df(x)}{dx} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}. \tag{1}$$

For the three points in consideration, we could write a *first-order approximation* using two points at each time, resulting on the two *one-sided differences*:

$$\left.\frac{df(x)}{dx}\right|_{x_i} \sim \frac{f(x_{i+2}) - f(x_i)}{2\Delta x},$$

and

$$\left.\frac{df(x)}{dx}\right|_{x_i} \sim \frac{f(x_{i+1}) - f(x_i)}{\Delta x}.$$

However, these approximations do not give us any instruction of how to relate together all the information that the function $f(x)$ at the three points carries (*i.e.,* this is a 3-point *stencil*[1]). In other words, the higher accuracy comes from the addition of one more point to the simple *two-points difference*, and it is clear that in the equations above the first approximation loses the information contained in the function at the point x_{x+2} and the second approximation loses the information that the function at x_{x+1} carries.

To evaluate a *second-order* accurate derivative with three points, we can use the definition of the *Taylor expansion* for a function $f(x)$ that is infinitely differentiable in a neighborhood of a point h:

$$f(x+h) = f(x) + f'(x)h + \frac{f''(x)}{2}h^2 + \mathcal{O}(h^3). \tag{2}$$

For the three points in consideration, the Eq. 2 becomes

$$f(x_{i+2}) = f(x_i + 2\Delta x) = f(x_i) + f'(x_i)(2\Delta x) + \frac{f''(x_i)}{2}(2\Delta x)^2 + \mathcal{O}((2\Delta x)^3),$$

and

$$f(x_{i+1}) = f(x_i + \Delta x) = f(x_i) + f'(x_i)\Delta x + \frac{f''(x_i)}{2}\Delta x^2 + \mathcal{O}(\Delta x^3).$$

We can now set a *computed derivative*, $f'(x_i)$, by rewriting the equations above as

$$f'(x_i) = \frac{f(x_{i+2}) - f(x_i)}{2\Delta x} - \frac{f''(x_i)}{2}(2\Delta x) + \mathcal{O}((2\Delta x)^2).$$

[1] The range of points involved is called the stencil.

and

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{\Delta x} - \frac{f''(x_i)}{2}(\Delta x) + \mathcal{O}(\Delta x^2).$$

Rewriting the above (second) equation in a more suitable way (multiplying both sides by $\times[-2]$ and the first term in the right side by $\times[2/2]$),

$$-2f'(x_i) = \frac{-4f(x_{i+1}) + 4f(x_i)}{2\Delta x} + \frac{f''(x_i)}{2}(2\Delta x) + (-)\mathcal{O}(2\Delta x^2),$$

and adding both the relations,

$$-f'(x_i) = \frac{f(x_{i+2}) - f(x_i) - 4f(x_{i+1}) + 4f(x_i)}{2\Delta x} + (2)\mathcal{O}(\Delta x^2),$$

results on the *second order accurate first derivative* at x_i:

$$\boxed{f'(x_i) = \frac{-f(x_{i+2}) + 4f(x_{i+1}) - 3f(x_i)}{2\Delta x} + \mathcal{O}(\Delta x^2).}$$

The last term in the above equation, $\mathcal{O}(\Delta x)^2$, is the grid size, denoting the (leading term in) truncation error (the order of accuracy in the error representation). Since this term gives an error estimation, any multiplicative constant is absorbed, *i.e.*, the order of the term is the only meaningful information. This fact can be seen when we represent the methods developed in this exercise for some analytic function (*e.g.*, $f(x) = \sin(x)$), as in the Fig. 1 (motivated by the same discussion in class).

(b) When deriving the *Simpson's rule*, we can obtain a quadratic function passing through $f(x)$ at the points x_0, x_1 and x_2,

$$f(x) = \frac{f_0 - 2f_1 + f_2}{2\Delta x^2}(x - x_0)^2 + \frac{-f_2 + 4f_1 - 3f_0}{2\Delta x}(x - x_1) + f_0. \tag{3}$$

Show that

(i) The derivative of $f(x)$ at point x_1 recovers the centered difference formula.

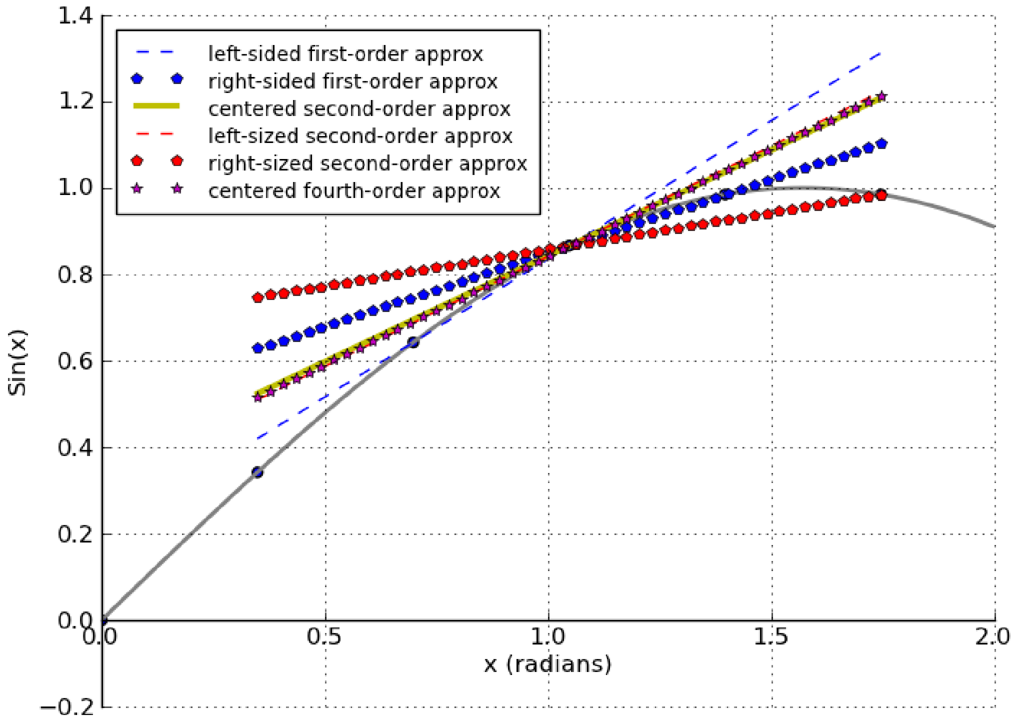

Figure 1: Comparison of the first derivative of $f(x) = \sin(x)$ at the point $x = 1$ to many orders of accuracy and to centered and one-sided (right and left) differences.

The derivative of Eq. 3 is

$$\frac{df(x)}{dx} = \left(\frac{f_0 - 2f_1 + f_2}{2\Delta x^2}\right)(2)(x - x_0) + \left(\frac{-f_2 + 4f_1 - 3f_0}{2\Delta x}\right), \quad (4)$$

which calculating at the point x_1 and making $\Delta x = x_1 - x_0$ gives the *second order centered difference formula*,

$$\boxed{f'(x_1) = \frac{f_2 - f_0}{2\Delta x}.}$$

(ii) `The derivative of` $f(x)$ `at` x_0 `gives the same expression from part (a), but now right-sided.`

When calculating the Eq. 4 at the point $x = x_0$, the first right-side term vanishes, resulting in a similar expression obtained in the item (a) (but right-sided if we say $x_0 \rightarrow x_{i+2}$),

$$\boxed{f'(x_0) = \frac{-f_2 + 4f_1 - 3f_0}{2\Delta x}.}$$

Q.2 **(Derivative Error Estimates) Starting with the** *second-order centered difference* **equation (as in the item Q.1-b-i),**

$$\Delta_1(h) = \frac{f(x+h) - f(x-h)}{2h}; \tag{5}$$

write a program to compute the numerical derivative of $f(x) = \sin(x)$ at $x = 1$. By comparing $\Delta_1(h)$ and $\Delta_1(h/2)$, reach *relative error* $\epsilon = 10^{-7}$**. In addition, return the** *Richardson* **extrapolated value of $f'(x)$, which is $\mathcal{O}(h^4)$.**

A real number x has a machine representation given by

$$fl(x) = x(1+\epsilon),$$

with a given precision $|\epsilon| \leq \epsilon_M$, where ϵ_M is the machine precision (*i.e.,* single:10^{-7}, double:10^{-16}). Therefore, we cannot represent all decimal numbers with an exact binary representation in a computer (*e.g.,* the binary representation of the decimal 0.1). This is called the *roundoff error.*

To numerically calculate derivatives of a function $f(x)$ to a set of data, we can interpolate the data to a uniform grid with $2 \leq i \leq n$ points, with convenient distances, h, between each of them. In this case, a *truncation error* is also introduced, so that the total *absolute* error is

$$\epsilon_{\text{total}} = \left| f'(x) - \frac{\bar{f}(x_{i+1}) - \bar{f}(x_i)}{\Delta x} \right| \leq \frac{|f''|h}{2} + \frac{2\epsilon_M}{h}, \tag{6}$$

with $\bar{f}(x_i)$ being an approximation of $f(x_i)$. The first term in the right-side is the truncation error, and the second term is the roundoff error.

When the function $f(x)$ is available analytically we can make estimates of error and control the *accuracy* in an adaptive scheme. With the Eq. 5, we can chose values for h that make the comparison of $\Delta_1(h)$ to $\Delta_1(h/2)$ close to the desired error, which in this case is $\epsilon_{\text{desired}} \sim 10^{-7}$. This is the same order of a single precision machine error, so that calculating our machine error gives $\epsilon_{\text{roundoff}} \sim 2.22 \times 10^{-16} \ll \epsilon_{\text{desired}}$.

To estimate the first derivative (and respective errors) we iteratively build more accurate approximations, adding to our approximation of $f'(x)$ higher order of Taylor expansion terms. These higher order terms can be calculated at h and $h/2$ (*i.e.,* in between grid intervals), resulting on the *Richardson extrapolation* of the value of f',

$$f'_{\text{R}} = -\frac{\Delta_1(h) - 4\Delta_1(h/2)}{3} + \mathcal{O}(h^4).$$

The errors calculated in this approximation can be *absolute*,

$$\left| f'(x) - f'_{\mathrm{R}} \right| \leq \epsilon_{\mathrm{a}},$$

relative,

$$\left| \frac{f'(x) - f'_{\mathrm{R}}}{f'(x)} \right| \leq \epsilon_{\mathrm{r}},$$

and a *truncation* error $\mathcal{O}(h^4)$,

$$h^2 \times \left| \Delta_1(h) - \Delta_1(h/2) \right| \leq \epsilon_{\mathrm{t}} \sim \epsilon_{\mathrm{desired}}.$$

For the derivative of $f(x) = \sin(x)$ at $x = 1$, Fig. 2 shows a comparison among these three errors versus the grinding size (h) and the iteration size (n).

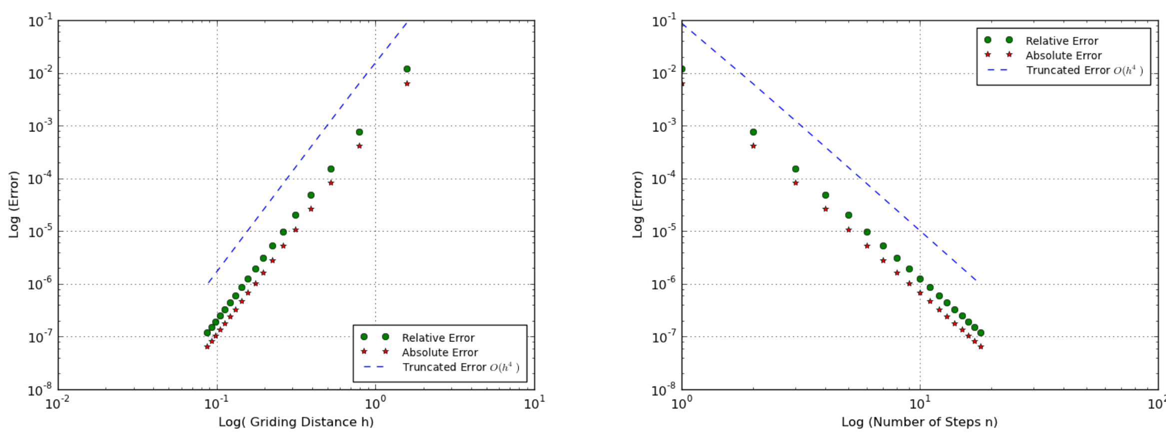

Figure 2: (left) Absolute, relative, and truncation errors versus the grinding size h, for the calculation of the first order derivative of $f(x) = \sin(x)$ at $x = 1$ by Richardson extrapolation method. (right) Same calculations but for the number of iterations n.

Q.3 (Simpson's Rule) In class we derived the compound version of the Simpson's rule, which we integrate over pairs of slabs/intervals:

We want to solve numerically the integral

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

The simplest case is by a *piecewise constant interpolation*, *i.e.,* the *midpoint rule*,

$$I_{\rm m} \sim (b-a) f\Big(\frac{a+b}{2}\Big).$$

A little better approximation is by a *piecewise linear interpolation*, *i.e.,* the *trapezoid rule*,

$$I_{\rm t} \sim (b-a)\frac{f(b)+f(a)}{2},$$

The accuracy is better for higher-order interpolating polynomials. If we try to approximate the integral by a parabola, with $\delta = (b-a)/2$,

$$f(x) = A(x-x_0)^2 + B(x-x_1) + C, \tag{7}$$

with

$$A = \frac{f_0 - 2f_1 + f_2}{2\delta^2},$$

$$B = -\frac{f_2 - 4f_1 + 3f_0}{2\delta},$$

and

$$C = f_0,$$

we derive the *Simpson's rule*,

$$\begin{aligned} I_{\rm S} &\sim \int_{x_0}^{x_2} \Big[A(x-x_0)^2 + B(x-x_0) + C\Big]dx, \\ &\sim \frac{\delta}{3}\Big(f_0 + 4f_1 + f_2\Big). \end{aligned}$$

These simple integration methods are not accurate for a larger domain, *i.e.,* when $[a, b]$ is large. In principle, we could keep adding higher order polynomials terms to get more accuracy. However, a better options is called *compound integration*, which breaks the domain into sub-domains and use the integration rules in each of the respective slabs,

$$I = \int_a^b f(x)dx = \sum_{i=0}^{N-1} \int_{x_i}^{x_i+1} f(x)dx.$$

The *compound Simpson's* integral method is given by

$$\boxed{I_{\text{cS}} = \frac{h}{3} \sum_{i=0}^{N/2-1} \Big(f_{2i} + 4 f_{2i+1} + f_{2i+2} \Big) + \mathcal{O}(h^4),}$$

which in order to pair up all the slices, we have to have an *even* number of slices.

(a) `Imagine that you want to integrate` $f(x)$ `over` $[a, b]$`, and have divided the domain into an odd number,` N`, slabs/intervals, with the function specified at the points` $x_0, ..., x_N$`. In this case, you would integrate all the pairs of slabs up until the last slab. For the remaining odd slab,` $[x_{N-1}, x_N]$`, show that a Simpson's rule for the slab is`

$$\int_{x_{N-1}}^{x_N} f(x)dx \sim \frac{h}{12}(-f_{N-2} + 8 f_{N-1} + 5 f_N). \tag{8}$$

To derive Eq. 8 we first fit it to a parabola to the last three points, as in Eq. 7, where $x_0 = x_{N-2}, x_1 = x_{N-1}, x_2 = x_N$,

$$f_{\text{cS}}(x) = A(x - x_{N-2})^2 + B(x - x_{N-2}) + C,$$

with

$$A = \frac{f_{N-2} - 2 f_{N-1} + f_N}{2\delta^2},$$

$$B = -\frac{f_N - 4 f_{N-1} + 3 f_{N-2}}{2\delta},$$

and

$$C = f_{N-2}.$$

Integrating over the last slab, with $\delta \to h = x_{i+1} - x_i$, gives the desired result,

$$
\begin{aligned}
\int_{b-h}^{b} f(x)dx &= \int_{x_{N-1}}^{x_N} \Big[A(x - x_{N-2})^2 + B(x - x_{N-2}) + C\Big]dx, \\
&= \frac{A}{3}(x - x_{N-2})^3\Big|_{x_{N-1}}^{x_N} + \frac{B}{2}(x - x_{N-2})^2\Big|_{x_{N-1}}^{x_N} + Cx\Big|_{x_{N-1}}^{x_N}, \\
&= \frac{f_{N-2} - 2f_{N-1} + f_N}{6\delta^2}(x - x_{N-2})^3\Big|_{x_{N-1}}^{x_N} - \frac{f_N - 4f_{N-1} + 3f_{N-2}}{4\delta}(x - x_{N-2})^2\Big|_{x_{N-1}}^{x_N} \\
&\quad + f_{N-2}x\Big|_{x_{N-1}}^{x_N}, \\
&= \frac{f_{N-2} - 2f_{N-1} + f_N}{6h^2}\Bigg[(x_N - x_{N-2})^3 - (x_{N-1} - x_{N-2})^3\Bigg] \\
&\quad - \frac{f_N - 4f_{N-1} + 3f_{N-2}}{4h}\Bigg[(x_N - x_{N-2})^2 - (x_{N-1} - x_{N-2})^2\Bigg] \\
&\quad + f_{N-2}(x_N - x_{N-1}), \\
&= \frac{f_{N-2} - 2f_{N-1} + f_N}{6h^2}[7h^3] - \frac{f_N - 4f_{N-1} + 3f_{N-2}}{4h}[3h^2] + f_{N-2}[h], \\
&= \frac{7f_{N-2} - 14f_{N-1} + 7f_N}{6}h - \frac{3f_N - 12f_{N-1} + 9f_{N-2}}{4}h + f_{N-2}h, \\
&= \frac{14f_{N-2} - 28f_{N-1} + 14f_N}{12}h - \frac{9f_N - 36f_{N-1} + 27f_{N-2}}{12}h + f_{N-2}h, \\
&= \frac{h}{12}\Big[(14 - 27 + 12)f_{N-2} + (-28 + 36)f_{N-1} + (14 - 9)f_N\Big], \\
&= \frac{h}{12}\Big[- f_{N-2} + 8f_{N-1} + 5f_N\Big].
\end{aligned}
$$

(b) Integration of $f(x) = \sin(\pi x)$ over $[0, 1]$ using $N = 3, 7, 15, 31$ slabs/intervals, with a plot with the absolute error vs. $\delta = (b - a)/N$ (Fig 3).

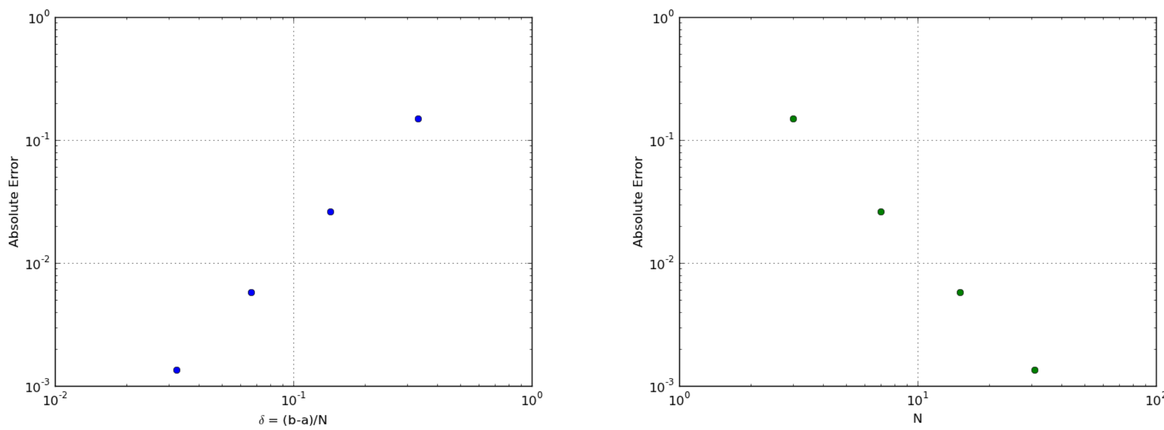

Figure 3: (left) Log-log plot of the absolute errors vs. $\delta = (b-a)/N$ showing a $\mathcal{O}(\delta^4)$ *convergence* in the calculation of the (Simpson compound) integration of $f(x) = \sin(\pi x)$ over $[0, 1]$ using $N = 3, 7, 15, 31$ slabs/intervals. (right) Same for absolute error vs. N.

Q.4 **(Gaussian Quadrature) In the Gauss-Legendre quadrature method, with n quadrature points, an exact integral of a polynomial up to degree $d = 2n-1$ is obtained.**

(1.) Consider a 5-point quadrature, with tabled roots and weights, for the Gauss-Legendre method. Compute

$$I = \int_0^1 p(x)dx,$$

where $p(x)$ is a 9th degree polynomial

$$p(x) = \sum_{k=0}^{9} x^k.$$

(2.) Compute also the compound version of the Simpson's rule (2 pairs of intervals) and the (3.)error against the exact integral.

In the Gaussian quadrature method, instead of a fixed spacing, we express the integral as a sums of weighted pieces, *i.e.*, we choose the location of each x_i. The fundamental theorem states that for some polynomial $q(x)$ of degree N, such that

$$\int_a^b q(x)\rho(x)x^k dx = 0,$$

such that $k = 0, .., N-1$ and $\rho(x)$ is a weight function. We pick the points $x_1, x_2, ...x_N$ as the roots of the polynomial $q(x)$. For a sets of weights $\omega_1, ..., \omega_N$, the integral

$$\int_a^b f(x)\rho(x)dx \sim \omega_1 f(x_1) + ... + \omega_N f(x_N),$$

is exact if $f(x)$ is a polynomial of degree $< 2N$. This is more acurate than just fitting the function to a polynomial of degree $N-1$, as when we had fixed grid of N points.

References

[1] *Mike Zingale's Class*, *http://bender.astro.sunysb.edu/classes/phy688_spring2013*

[2] *An Introduction to Computational Physics*, T. Pang, Cambridge Press

[3] *Computational Physics*, M. Hjorth-Jensen