

# NOTES ON ERROR ANALYSIS OF NUMERICAL INTEGRATION

For your homework, you'll be doing error analysis in the simplest way possible. I've asked you to integrate a function that you can integrate with pen and paper, so to determine the dependence of the error  $E$  on the stepsize  $h$  all you need to do is to compute (numeric answer - analytic answer) for each value of  $h$ .

The Taylor series business is a way to *predict* what these errors will be before you test them, and to determine how good an algorithm will be. You don't necessarily need to reproduce this, but you should understand it.

## 1 The approach

We want to come up with a formula for each of our three methods that relates the error to the stepsize for a single bin. Each of the integration methods approximates the shape of the function as some shape (a rectangle or a trapezoid); we will write down the functional form of that shape and integrate it, and then subtract the "actual" integral of the function.

If we are doing this for arbitrary functions, we don't know the actual integral. This is where the Taylor series comes in: we can write a Taylor series that expands the function about the left-hand side of the Riemann sum bin (which I shall call  $x=0$ ). This is convenient, since we already have a "small quantity" in mind: the bin width  $h$ . We will then get a polynomial in  $h$  for both the exact integral and the approximation, which will make it very easy to compare terms and figure out what power of  $h$  appears in the leading term in the error. Why do we care about the power? Simply, because if  $h$  is small enough, the power of  $h$  outweighs any coefficient in front.

## 2 Integrating the exact function

Call the function we'd like to integrate  $f(x)$ ; let's integrate it over a single bin. Its Taylor series up to second order is

$$f(x) = f(0) + xf'(0) + \frac{1}{2}x^2f''(0) + \mathcal{O}(x^3) \quad (1)$$

Its integral is then

$$\int_0^h f(x) dx = \int_0^h \left[ f(0) + xf'(0) + \frac{1}{2}x^2f''(0) + \mathcal{O}(x^3) \right] dx \quad (2)$$

$$= xf(0) + \frac{1}{2}x^2f'(0) + \frac{1}{2}\frac{1}{3}x^3f''(0) + \mathcal{O}(x^4) \Big|_0^h \quad (3)$$

$$= hf(0) + \frac{1}{2}h^2f'(0) + \frac{1}{6}h^3f''(0) + \mathcal{O}(h^4) \quad (4)$$

This last equation is, then, the integral of  $f(x)$  over one bin. We will be subtracting the results we get from integrating the approximations from it to see how they compare.

### 3 The left-hand rule

The left-hand rule approximates the function as a horizontal line of height equal to the function's value at the left side of the interval:

$$f_{\text{LHR}}(x) = f(0) \tag{5}$$

This is trivially easy to integrate:

$$\int_0^h f_{\text{LHR}}(x) dx = \int_0^h f(0) dx \tag{6}$$

$$= x f(0) \Big|_0^h \tag{7}$$

$$= h f(0) \tag{8}$$

Subtracting this from the exact result, we see we've gotten the first term right, but all the rest are wrong. Since we only really care about the leading-order error term, we can write the error as

$$E_{\text{LHR}} = \frac{1}{2} h^2 f(0) + \mathcal{O}(h^3). \tag{9}$$

Thus we see that the error made by the left-hand rule is of order  $h^2$ . This is the error *per bin*; if we want the error for the whole integral, we have to multiply by the number of bins  $N_b = \frac{b-a}{h}$ . The power of  $h$  in the denominator cancels one of the powers of  $h$  in the error per bin. This makes sense: if you cut the bin width in half, you require twice as many bins, so you will get one-half, not one-fourth, the overall error. Thus, we conclude:

- The left-hand rule is a **first-order** algorithm: the error in the integral is proportional to  $h$ .

### 4 The midpoint rule

The midpoint rule likewise approximates the function as a horizontal line, but this time one evaluated at the midpoint of the interval. In order to know how high this line is, we have to evaluate the function at its midpoint. We could write  $f(h/2)$ , but this is unhelpful, since everything else in our analysis involves the evaluation of  $f$  and its derivatives at the left-hand side of the interval. So we instead write  $f(h/2)$  in terms of the Taylor series:

$$f_{\text{mid}}(x) = f(0) + \frac{1}{2} h f'(0) + \frac{1}{2} \left(\frac{1}{2}\right)^2 h^2 f''(0) \tag{10}$$

As with the Left-Hand Rule, we can trivially integrate this by multiplying by  $h$ , since it's just a rectangle with the height given above and width  $h$ . This gives

$$\int_0^h f_{\text{mid}}(x)dx = \int_0^h \left[ f(0) + \frac{1}{2}hf'(0) + \frac{1}{8}h^2f''(0) \right] dx \quad (11)$$

$$= hf(0) + \frac{1}{2}h^2f'(0) + \frac{1}{8}h^3f''(0) \quad (12)$$

Note here that the first two terms match the exact integral form (Eq. 4). The approximation made by the midpoint rule correctly “captures” both the height of the function at the left-hand side *and* its slope – the first two terms of the full Taylor series, not just the first one. The error is thus

$$E_{\text{mid}} = \frac{1}{24}h^3f''(0) + \mathcal{O}(h^4). \quad (13)$$

(The  $1/24$  comes from subtracting  $1/8$  from  $1/6$ .) Since the leading term in the “error per bin” formula is  $\mathcal{O}(h^3)$ , the logic above says that the overall error in the integral is proportional to  $h^2$ .

Thus:

- The midpoint rule is a **second-order** algorithm: the error in the integral is proportional to  $h^2$ .

**Note the significance of this. In the left-hand rule, if we cut the stepsize by a factor of ten, we improve the accuracy by roughly a factor of ten (plus higher order corrections), since it’s a first-order algorithm. However, in the midpoint rule, if we cut the stepsize by a factor of ten, we improve the accuracy by a factor of a hundred. This is why second-order algorithms are so much better!**

## 5 The trapezoid rule

The trapezoid rule approximates the function as a sloped line whose y-intercept is  $f(0)$  and whose slope is  $(f(h) - f(0))/h$ . As with the midpoint rule, we write  $f(h)$  in terms of the Taylor expansion about  $f(0)$ . So the slope and intercept of the line are:

$$m = \frac{[f(0) + hf'(0) + \frac{1}{2}h^2f''(0)] - f(0)}{h} = f'(0) + \frac{1}{2}hf''(0) \quad (14)$$

$$b = f(0) \quad (15)$$

giving us a slope-intercept equation of

$$f_{\text{trap}} = f'(0)x + \frac{1}{2}hf''(0)x^2 \quad (16)$$

which we can integrate to get

$$\int_0^h f_{\text{trap}}(x) dx = \int_0^h \left[ f'(0)x + \frac{1}{2}hx f''(0) \right] dx \quad (17)$$

$$= hf(0) + \frac{1}{2}h^2 f'(0) + \frac{1}{4}h^3 f''(0) \quad (18)$$

giving an error of

$$E_{\text{trap}} = -\frac{1}{12}h^3 f''(0) + \mathcal{O}(h^4). \quad (19)$$

(As before,  $1/12$  comes from subtracting  $1/4$  in the trapezoid approximation formula from  $1/6$  in the “correct” answer in Eq. 4.) As with the Midpoint Rule, we have gotten the first two terms right; the error is of order  $h^3$  for one bin, or order  $h^2$  for the whole integral. Thus:

- The trapezoid rule is also a **second-order** algorithm: the error in the integral is proportional to  $h^2$ .

## 6 Summary

We have four formulas, one for the exact integral, and the three approximations:

$$\int_0^h f_{\text{exact}}(x) dx = hf(0) + \frac{1}{2}h^2 f'(0) + \frac{1}{6}h^3 f''(0) + \mathcal{O}(h^4) \quad \int_0^h f_{\text{LHR}}(x) dx = hf(0) \quad (20)$$

$$\int_0^h f_{\text{mid}}(x) dx = hf(0) + \frac{1}{2}h^2 f'(0) + \frac{1}{8}h^3 f''(0) \quad (21)$$

$$\int_0^h f_{\text{trap}}(x) dx = hf(0) + \frac{1}{2}h^2 f'(0) + \frac{1}{4}h^3 f''(0) \quad (22)$$

The left-hand rule is a first-order algorithm; the other two are second-order algorithms.

## 7 Simpson’s rule

You might note here that we have two second-order algorithms that have *different* incorrect values for the  $\mathcal{O}(h^3)$  term. By taking a weighted average of the two of these, we can make the  $\mathcal{O}(h^3)$  term come out right as well. This approach is called *Simpson’s rule*.

$$f_{\text{Simpson}}(x) = \frac{2}{3}f_{\text{mid}}(x) + \frac{1}{3}f_{\text{trap}}(x) \quad (23)$$

Then the integral is

$$\int_0^h f_{\text{Simpson}}(x) dx = hf(0) + \frac{1}{2}h^2 f'(0) + \frac{1}{6}h^3 f''(0) \quad (24)$$

which matches what we were “supposed to get”, except for errors of order  $\mathcal{O}(h^4)$  that we haven’t calculated. (To do this calculation, you’ll need to carry all the Taylor series involved out to one more order. Anyone who does this calculation and can explain it to me will get substantial extra credit.)

So what is the coefficient of the  $\mathcal{O}(h^4)$  error term in Simpson’s rule? When you do this – carry all the Taylor series out to one more order – you’ll see that by fortunate coincidence the  $\mathcal{O}(h^4)$  error is zero, as well; the leading order in Simpson’s rule is actually of order  $\mathcal{O}(h^5)$ ! This means that:

- Simpson’s rule is a fourth-order integration algorithm.

In our class, the midpoint or trapezoid rules will be enough. In principle, you could devise ever more complex integration algorithms that are fifth, sixth, etc., order, but quite rapidly you wind up exhausting the numerical precision of your computing device; computers are fast enough that Simpson’s rule is generally fine for any integral you’d want to do with Riemann summation.