Lecture 7

We can perform an analogous error analysis for the Simpson rule.

Approx. error to leading order is

$$\varepsilon = \frac{1}{90} h^4 \left[f'''(a) - f'''(b) \right]$$

So Simpson's rule is 3rd order integration rule w/ 4th order approximation error.

rounding error for Simpson's rule is of order $C \int_a^b f(x)\,dx$

so that

$$N = (b-a) \sqrt[4]{\frac{f'''(a) - f'''(b)}{90 \int_a^b f(x)\,dx}} \; C^{-1/4}$$

so rounding error becomes important roughly when $N \cong 10{,}000$

No point using larger values of N w/ Simpson's rule b/c you approach machine precision

Simpson's rule may not always be better than trapezoid rule. Function might be such that $f'''(a)$ is very large & make Simpson error larger than trap. error

---

In many cases, we might not be able to compute derivatives or the function we're trying to integrate might not be a function but instead be experimental data.
How to estimate error in this case?

Can use a trick..

Suppose we're using trap. rule for $x=a$ to $x=b$ & suppose # of steps is N_1 & step size is

$$h_1 = \frac{b-a}{N_1}$$

Trick is to double the number of steps & do the integral again.

I.e., define $N_2 = 2N_1$ &

$$h_2 = \frac{(b-a)}{N_2} = \tfrac{1}{2}h_1$$

Since trapezoidal rule has error $O(h^2)$, when we halve the slice size h, we quarter the size of error. Good payoff for increased complexity...

4

Suppose true value of integral is I. denote 1st estimate by I_1 (# of steps is N_1)

Difference between true value & estimate is $O(h^2)$, so that we can write

$$I - I_1 = ch_1^2$$

$$\Rightarrow \quad I = I_1 + ch_1^2$$ (neglecting higher order terms)

Can do the same for second estimate, writing

$$I = I_2 + ch_2^2$$

$$\Rightarrow$$

$$I_1 + ch_1^2 = I_2 + ch_2^2$$

Using that $h_2 = h_1/2$, we get

$$I_1 + c\,4h_2^2 = I_2 + ch_2^2$$

$$\Rightarrow \quad I_2 - I_1 = 3ch_2^2$$

So this implies that the error on the 2^{nd} estimate is

$$\varepsilon_2 = c h_2^2 = \frac{1}{3}(I_2 - I_1)$$

So we get a simple way to estimate errors while avoiding the Euler-Maclaurin formula.

Can use the same idea to get an estimate for Simpson's rule of

$$\varepsilon_2 = \frac{1}{15}(I_2 - I_1)$$

---

choosing the number of steps for an integral.

often we'd like to calculate to some desired accuracy & we'd like to know how many steps are needed.

Would like to use as many steps needed to get to machine precision

Simple method to do this:

Start w/ given N &

keep doubling until reaching

desired accuracy.

Can use formula $\varepsilon_2 = \frac{1}{3}(I_2 - I_1)$

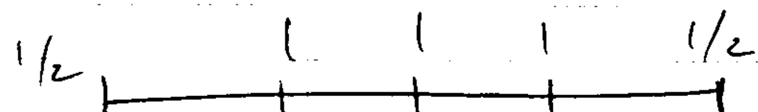
to figure out error.

error on i^{th} step is given by

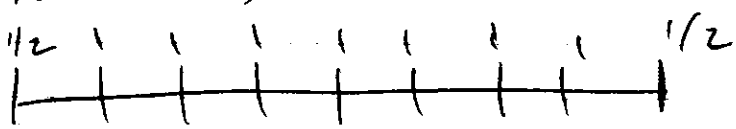
$$\varepsilon_i = \frac{1}{3}\left(I_i - I_{i-1}\right)$$

Nice feature: when doubling number of steps no need to recompute, just sample more & add

For trap. rule, initial sample points & coefficients are given by



new ones are then



sample points for first estimate are
nested inside those for the second estimate.

To make this precise, consider i^{th} iteration of trap. rule.
Let N_i be # of slices &
$h_i = \frac{b-a}{N_i}$ be width of each slice

Then $N_{i-1} = \frac{1}{2} N_i$ & $h_{i-1} = 2h_i$
&

$$I_i = h_i \left[\tfrac{1}{2} f(a) + \tfrac{1}{2} f(b) + \sum_{k=1}^{N_i - 1} f(a+kh_i) \right]$$

$$= h_i \Big[\tfrac{1}{2} f(a) + \tfrac{1}{2} f(b) + \sum_{\substack{k \text{ even} \\ 2,\dots,N_i-2}} f(a+kh_i)$$

$$+ \sum_{\substack{k \text{ odd} \\ 1,\dots,N_i-1}} f(a+kh_i) \Big]$$

Now consider that

$$\sum_{\substack{k \text{ even} \\ 2,\dots,N_i-2}} f(a+kh_i) = \sum_{k=1}^{N_i/2-1} f(a+2kh_i)$$

$$= \sum_{k=1}^{N_{i-1}-1} f(a+\cancel{2}kh_{i-1})$$

$$\Rightarrow \quad I_i = \tfrac{1}{2}h_{i-1}\left[\tfrac{1}{2}f(a) + \tfrac{1}{2}f(b) + \sum_{k=1}^{N_{i-1}} f(a+kh_{i-1})\right]$$

$$+ h_i \sum_{\substack{k \text{ odd} \\ 1,\dots,N_i-1}} f(a+kh_i)$$

$$\Rightarrow \quad I_i = \underbrace{\tfrac{1}{2} I_{i-1}}_{\substack{\text{old estimate} \\ \text{from } i-1 \\ \text{iteration}}} + \underbrace{h_i \sum_{\substack{k \text{ odd} \\ 1,\dots,N_i-1}} f(a+kh_i)}_{\text{new estimate}} \qquad (*)$$

This means we can compute adaptively & have nearly the same expense computationally had we proceeded nonadaptively

Algorithm:

1. Choose an initial # of steps N_1 & decide on desired target accuracy. Calculate 1st approx. I_1 using N_1 & trap. rule. set $i=2$.

2. Double the number of steps & use (*) to calculate I_i. calculate error using $\varepsilon_i = \frac{1}{3}(I_i - I_{i-1})$

3. If $|\varepsilon_i| <$ desired accuracy, stop. Otherwise $i=i+1$ & goto 2.

There is a similar formulation for the Simpson rule

Upon doubling the number of steps, we have

$$\varepsilon_i = \frac{1}{15}\left(I_i - I_{i-1}\right)$$

The iteration rule involves

$$S_i = \frac{1}{3}\left[f(a) + f(b) + 2\sum_{\substack{k \text{ even} \\ 2,\dots,N_i-2}} f(a+kh_i)\right]$$

$$T_i = \frac{2}{3}\sum_{\substack{k \text{ odd} \\ 1,\dots,N_i-1}} f(a+kh_i)$$

so that

$$S_i = S_{i-1} + T_{i-1}$$

$$I_i = h_i\left(S_i + 2T_i\right)$$

So the algorithm for the Simpson rule is

1. Choose an initial number of steps + a desired ~~target~~ accuracy.
   Calculate S_1, T_1, + I_1. Set $i=2$
2. Double the number of steps.
   Calculate S_i, T_i, + I_i
   Calculate error as
   $$\varepsilon_i = \frac{1}{15}(I_i - I_{i-1})$$
3. If $|\varepsilon_i| <$ desired target accuracy, stop. Else, set $i=i+1$ + go to 2.

Romberg integration

can do better than adaptive method of last section.

Recall that leading-order error for trap. rule is

$$ch_i^2 = \frac{1}{3}(I_i - I_{i-1})$$

True value of integral is

$$I = I_i + ch_i^2 + O(h_i^4)$$

where $O(h_i^4)$ is next term in series (recall that there are only even-order terms in series)

So

$$I = I_i + \frac{1}{3}(I_i - I_{i-1}) + O(h_i^4)$$

expression is now accurate to third order & has fourth-order error.

Can take this process further. Define

$R_{i,1} = I_i$

$R_{i,2} = I_i + \frac{1}{3}(I_i - I_{i-1}) = R_{i,1} + \frac{1}{3}(R_{i,1} - R_{i-1,1})$

Then from before, we have that

$$I = R_{i,2} + c_2 h_i^4 + O(h_i^6)$$

c_2 is some other constant & recall series has only even-order terms.

By the same reasoning, we have

$$I = R_{i-1,2} + c_2 h_{i-1}^4 + O(h_{i-1}^6)$$

$$= R_{i-1,2} + 16 c_2 h_i^4 + O(h_i^6)$$

equate the last two equations to get

$$c_2 h_i^4 = \frac{1}{15}\left(R_{i,2} - R_{i-1,2}\right) + O(h_i^6)$$

Substitute back in to get

$$I = R_{i,2} + \frac{1}{15}\left(R_{i,2} - R_{i-1,2}\right) + O(h_i^6)$$

good news is that we have eliminated the h_i^4 term & generated an estimate accurate to 5th order w/ sixth order error.

Can continue w/ this process, canceling out higher-order terms & getting more accurate results.

general rule is as follows:

Let $R_{i,m}$ be estimate calculated @ ith round of doubling procedure & accurate to order h^{2m-1} w/ error of order h^{2m}, then

$$I = R_{i,m} + c_m h_i^{2m} + O(h_i^{2m+2})$$

$$I = R_{i-1,m} + c_m h_{i-1}^{2m} + O(h_{i-1}^{2m+2})$$

$$= R_{i-1,m} + 4^m c_m h_i^{2m} + O(h_i^{2m+2})$$

Equating & rearranging gives

$$c_m h_i^{2m} = \frac{1}{4^m - 1}(R_{i,m} - R_{i-1,m}) + O(h_i^{2m+2})$$

Substituting gives the estimate

$$I = R_{i,m+1} + O(h_i^{2m+2})$$

where

$$R_{i,m+1} = R_{i,m} + \frac{1}{4^m - 1}\left(R_{i,m} - R_{i-1,m}\right)$$

accurate to order h^{2m+1} w/ error of order h^{2m+2}.

error is

$$c_m h_i^{2m} = \frac{1}{4^m - 1}\left(R_{i,m} - R_{i-1,m}\right) + \text{H.O.}$$

Algorithm proceeds according to the following procedure:

$$\begin{array}{lllllllll}
I_1 = R_{1,1} & \searrow & & & & & \\
I_2 = R_{2,1} & \rightarrow & R_{2,2} & \searrow & & & \\
 & \searrow & & & & & \\
I_3 = R_{3,1} & \rightarrow & R_{3,2} & \rightarrow & R_{3,3} & & \\
 & \searrow & & \searrow & & \searrow & \\
I_4 = R_{4,1} & \rightarrow & R_{4,2} & \rightarrow & R_{4,3} & \rightarrow & R_{4,4}
\end{array}$$

In words

1. Calculate

$I_1 = R_{1,1}$,

$I_2 = R_{2,1}$, using trap. rule

2. Calculate $R_{2,2}$

3. Calculate

$I_3 = R_{3,1}$ using trap. rule

then get $R_{3,2}$ & $R_{3,3}$

4. At each stage, compute

$I_i = R_{i,1}$ & then

$R_{i,2}, \ldots, R_{i,i}$

5. can also estimate error

Method is called Romberg integration

complexity is essentially the same as trap. rule while being much more accurate.