Integration

Adaptive Quadrature

Ashley Gannon

Fall 2020

Adaptive Quadrature

Adaptive Quadrature

While Romberg integration is more efficient than the composite Simpson's 1/3 rule, both methods use equispaced points.

Adaptive Quadrature

While Romberg integration is more efficient than the composite Simpson's 1/3 rule, both methods use equispaced points.

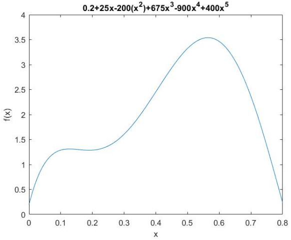
This constraint does not take into account that some functions have regions of relatively abrupt changes where more refined spacing might be required.

Adaptive Quadrature

While Romberg integration is more efficient than the composite Simpson's 1/3 rule, both methods use equispaced points.

This constraint does not take into account that some functions have regions of relatively abrupt changes where more refined spacing might be required.

- These methods require that fine spacing be applied everywhere on the domain even though it is only needed for the regions of sharp change

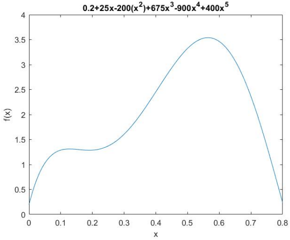


Adaptive Quadrature

While Romberg integration is more efficient than the composite Simpson's 1/3 rule, both methods use equispaced points.

This constraint does not take into account that some functions have regions of relatively abrupt changes where more refined spacing might be required.

- These methods require that fine spacing be applied everywhere on the domain even though it is only needed for the regions of sharp change



Adaptive quadrature methods remedy this situation by automatically adjusting the step size so that small steps are taken in regions of sharp variations and larger steps are taken where the function changes gradually.

Adaptive Quadrature

Adaptive quadrature methods accommodate the fact that many functions have regions of high variability along with other sections where change is gradual.

Adaptive Quadrature

Adaptive quadrature methods accommodate the fact that many functions have regions of high variability along with other sections where change is gradual.

- They accomplish this by adjusting the step size so that small intervals are used in regions of rapid variations and larger intervals are used where the function changes gradually.

Adaptive Quadrature

Adaptive quadrature methods accommodate the fact that many functions have regions of high variability along with other sections where change is gradual.

- They accomplish this by adjusting the step size so that small intervals are used in regions of rapid variations and larger intervals are used where the function changes gradually.
- Many of these techniques are based on applying the composite Simpson's 1/3 rule to subintervals in a fashion that is very similar to the way in which the composite trapezoidal rule was used in Richardson extrapolation.

Adaptive Quadrature

Adaptive quadrature methods accommodate the fact that many functions have regions of high variability along with other sections where change is gradual.

- They accomplish this by adjusting the step size so that small intervals are used in regions of rapid variations and larger intervals are used where the function changes gradually.
- Many of these techniques are based on applying the composite Simpson's 1/3 rule to subintervals in a fashion that is very similar to the way in which the composite trapezoidal rule was used in Richardson extrapolation.
    - The 1/3 rule is applied at two levels of refinement, and the difference between these two levels is used to estimate the truncation error.

Adaptive Quadrature

Adaptive quadrature methods accommodate the fact that many functions have regions of high variability along with other sections where change is gradual.

- They accomplish this by adjusting the step size so that small intervals are used in regions of rapid variations and larger intervals are used where the function changes gradually.
- Many of these techniques are based on applying the composite Simpson's 1/3 rule to subintervals in a fashion that is very similar to the way in which the composite trapezoidal rule was used in Richardson extrapolation.
  - The 1/3 rule is applied at two levels of refinement, and the difference between these two levels is used to estimate the truncation error.
  - If the truncation error is acceptable, no further refinement is required, and the integral estimate for the subinterval is deemed acceptable.

Adaptive Quadrature

Adaptive quadrature methods accommodate the fact that many functions have regions of high variability along with other sections where change is gradual.

- They accomplish this by adjusting the step size so that small intervals are used in regions of rapid variations and larger intervals are used where the function changes gradually.
- Many of these techniques are based on applying the composite Simpson's 1/3 rule to subintervals in a fashion that is very similar to the way in which the composite trapezoidal rule was used in Richardson extrapolation.
  - The 1/3 rule is applied at two levels of refinement, and the difference between these two levels is used to estimate the truncation error.
  - If the truncation error is acceptable, no further refinement is required, and the integral estimate for the subinterval is deemed acceptable.
  - If the error estimate is too large, the step size is refined and the process repeated until the error falls to acceptable levels

Adaptive Quadrature

Adaptive quadrature methods accommodate the fact that many functions have regions of high variability along with other sections where change is gradual.

- They accomplish this by adjusting the step size so that small intervals are used in regions of rapid variations and larger intervals are used where the function changes gradually.
- Many of these techniques are based on applying the composite Simpson's 1/3 rule to subintervals in a fashion that is very similar to the way in which the composite trapezoidal rule was used in Richardson extrapolation.
  - The 1/3 rule is applied at two levels of refinement, and the difference between these two levels is used to estimate the truncation error.
  - If the truncation error is acceptable, no further refinement is required, and the integral estimate for the subinterval is deemed acceptable.
  - If the error estimate is too large, the step size is refined and the process repeated until the error falls to acceptable levels
  - The total integral is then computed as the summation of the integral estimates for the subintervals.

Adaptive Quadrature

In theory, if we have the interval $x \in [a, b]$ with a width of $h_1 = b - a$, a first estimate with Simpson's 1/3 rule can be computed using

$$I(h_1) = \frac{h_1}{6}(f(a) + 4f(c) + f(b))$$

where $c = (a + b)/2$.

Adaptive Quadrature

In theory, if we have the interval $x \in [a, b]$ with a width of $h_1 = b - a$, a first estimate with Simpson's 1/3 rule can be computed using

$$I(h_1) = \frac{h_1}{6}(f(a) + 4f(c) + f(b))$$

where $c = (a + b)/2$.

As we did with Richardson Extrapolation, a more refined estimate can be computed by halving the interval, $h_2 = h_1/2 = (b - a)/2$

$$I(h_2) = \frac{h_2}{6}(f(a) + 4f(d) + 2f(c) + 4f(e) + f(b))$$

where $d = (a + c)/2$ and $e = (c + b)/2$.

Adaptive Quadrature

In theory, if we have the interval $x \in [a, b]$ with a width of $h_1 = b - a$, a first estimate with Simpson's 1/3 rule can be computed using

$$I(h_1) = \frac{h_1}{6}(f(a) + 4f(c) + f(b))$$

where $c = (a + b)/2$.

As we did with Richardson Extrapolation, a more refined estimate can be computed by halving the interval, $h_2 = h_1/2 = (b - a)/2$

$$I(h_2) = \frac{h_2}{6}(f(a) + 4f(d) + 2f(c) + 4f(e) + f(b))$$

where $d = (a + c)/2$ and $e = (c + b)/2$. Because both $I(h_1)$ and $I(h_2)$ are both estimates of the same integral, their difference can be used to measure the error of approximation

$$E \cong |I(h_2) - I(h_1)|$$

Adaptive Quadrature

In addition, the estimate and error associated with either application can be represented generally as

$$I = I(h_n) + E(h_n)$$

where I is the exact integral, $I(h_n)$ is the approximate integral, $E(h_n)$ is the truncation error.

Adaptive Quadrature

In addition, the estimate and error associated with either application can be represented generally as

$$I = I(h_n) + E(h_n)$$

where I is the exact integral, $I(h_n)$ is the approximate integral, $E(h_n)$ is the truncation error.

Using a similar approach to what we did for Richardson extrapolation, we can derive an estimate in the error of the more refined estimate $I(h_2)$ as a function of the difference between the two integral estimates so that

$$E(h_2) = \frac{1}{15}\left(I(h_2) - I(h_1)\right)$$

Adaptive Quadrature

In addition, the estimate and error associated with either application can be represented generally as

$$I = I(h_n) + E(h_n)$$

where I is the exact integral, $I(h_n)$ is the approximate integral, $E(h_n)$ is the truncation error.

Using a similar approach to what we did for Richardson extrapolation, we can derive an estimate in the error of the more refined estimate $I(h_2)$ as a function of the difference between the two integral estimates so that

$$E(h_2) = \frac{1}{15}\left(I(h_2) - I(h_1)\right)$$

Which can be substituted into the general formula above,

$$I = I(h_2) + \frac{1}{15}\left(I(h_2) - I(h_1)\right)$$

This formula is known as *Boole's rule*.

Adaptive Quadrature

These equations can be combined into an efficient algorithm.

Adaptive Quadrature

These equations can be combined into an efficient algorithm.

- Evaluate the initial application of Simpson's 1/3 rule by computing the two integral estimates

$$I(h_1) = \frac{h_1}{6}(f(a) + 4f(c) + f(b))$$
$$I(h_2) = \frac{h_2}{6}(f(a) + 4f(d) + 2f(c) + 4f(e) + f(b))$$

Adaptive Quadrature

These equations can be combined into an efficient algorithm.

- Evaluate the initial application of Simpson's 1/3 rule by computing the two integral estimates

$$I(h_1) = \frac{h_1}{6}(f(a) + 4f(c) + f(b))$$
$$I(h_2) = \frac{h_2}{6}(f(a) + 4f(d) + 2f(c) + 4f(e) + f(b))$$

- Estimate the error using the absolute difference between the two integral estimates, $|f(h_2) - f(h_1)|$.

Adaptive Quadrature

These equations can be combined into an efficient algorithm.

- Evaluate the initial application of Simpson's 1/3 rule by computing the two integral estimates

$$I(h_1) = \frac{h_1}{6}(f(a) + 4f(c) + f(b))$$
$$I(h_2) = \frac{h_2}{6}(f(a) + 4f(d) + 2f(c) + 4f(e) + f(b))$$

- Estimate the error using the absolute difference between the two integral estimates, $|f(h_2) - f(h_1)|$.
- Depending on the error, one of two things will happen
  1. If the error is less than or equal to the tolerance, we estimate the integral using Boole's rule:
  $$I = I(h_2) + \frac{1}{15}(I(h_2) - I(h_1))$$
  2. If the error is larger than the tolerance, `AdaptiveQuadrature` is recursively called for each subinterval, $[a, c]$ and $[c, b]$.

Adaptive Quadrature

These equations can be combined into an efficient algorithm.

- Evaluate the initial application of Simpson's 1/3 rule by computing the two integral estimates

$$I(h_1) = \frac{h_1}{6}(f(a) + 4f(c) + f(b))$$
$$I(h_2) = \frac{h_2}{6}(f(a) + 4f(d) + 2f(c) + 4f(e) + f(b))$$

- Estimate the error using the absolute difference between the two integral estimates, $|f(h_2) - f(h_1)|$.
- Depending on the error, one of two things will happen
  1. If the error is less than or equal to the tolerance, we estimate the integral using Boole's rule:
  $$I = I(h_2) + \frac{1}{15}\left(I(h_2) - I(h_1)\right)$$
  2. If the error is larger than the tolerance, `AdaptiveQuadrature` is recursively called for each subinterval, $[a, c]$ and $[c, b]$.

The beauty of this algorithm is in the two recursive calls. This recursion allows us to keep subdividing the intervals until our tolerance is met.

Adaptive Quadrature Pseudocode

```
Define function that takes in a, b, and tol
Declare/define h1 = b-a
Declare/define h2 = h1/2
Declare/define midpoints:
  c = (a+b)/2
  d = (a+c)/2
  e = (c+b)/2
Declare/define I

Declare/define Ih1 and Ih2
  Ih1 = (h1/6)*(f(a)+4*f(c)+f(b))
  Ih2 = (h2/6)*(f(a)+4*f(d) +2*f(c)+ 4*f(e) +f(b))

Declare/define the error
error = abs(Ih2-Ih1)
if error > tol
    Boole's Rule:
    I = Ih2 + (1/15)*(Ih2-Ih1)
else
    declare/define Ia = AdaptiveQuadrature(a,c,tol)
    declare/define Ib = AdaptiveQuadrature(c,b,tol)
    I = Ia + Ib
```

Let's code it!

Let's try this out on our problem

$$f(x) = 0.2 + 25x - 200x^2 + 675x^3 - 900x^4 + 400x^5$$

from $a = 0$ to $b = 0.8$ with a tol of 1e-6.

If we've done it correctly, we should have the sum, $I = 1.6405333$