MANE 3351

Lecture 15

Classroom Management

Agenda

- Romberg Integration
- Lab 6, if not completed
- Homework 3 due today
- Homework 4 assigned (due 10/29/2025)

Calendar

Week	Monday Lecture	Wednesday Lecture
8	**10/20:** Simpson's Rule	**10/22:** Romberg Integration
9	**10/27:** Gaussian Quadrature	**10/29:** Numerical Differentiation (not on Test 2)
10	**11/3:** Linear Algebra	**11/5:** Test 2 (Root Finding and Numerical Integration)

Resources

Handouts

- Lecture 15 Slides
- Lecture 15 Marked Slides

Lecture 15

Today's topic is Romberg Integration

- Clever combination of trapezoid rule and Richardson's Extrapolation
- Highly accurate
- Cheney and Kincaid (2004)[1] show example output in the form of a lower triangle from Romberg integration

The *Romberg algorithm* produces a triangular array of numbers, all of which are numerical estimates of the definite integral $\int_a^b f(x)\,dx$. The array is denoted here by the notation

$$
\begin{array}{llllll}
R(0,0) & & & & & \\
R(1,0) & R(1,1) & & & & \\
R(2,0) & R(2,1) & R(2,2) & & & \\
R(3,0) & R(3,1) & R(3,2) & R(3,3) & & \\
\vdots & \vdots & \vdots & \vdots & \ddots & \\
R(n,0) & R(n,1) & R(n,2) & R(n,3) & \cdots & R(n,n)
\end{array}
$$

Best estimate

Romberg Integration Results

Step 1

The first step is to calculate $R(0,0)$

- $R(0,0)$ is the result of applying the Trapezoid rule with 1 interval
- $R(0,0) = \frac{1}{2}(b-a)[f(a)+f(b)]$
- For our example of the standard normal pdf with $a = -5$, and $b = 0$, we observe
  - $R(0,0) = \frac{1}{2}(0-(-5.0))[f(-5.0)+f(0.0)] = \frac{1}{2}(5.0)[0.0 + 0.398942] = 0.997355$
  - This is a very poor approximation to the true value of 0.5

Step 2

Start a second row and calculate $R(1,0)$. For each new row, double the number of intervals used in the trapezoid rule

- $R(1,0)$ is the trapezoid with two intervals
- The general formula for $R(n,0)$ is

$$R(n,0) = \frac{1}{2}R(n-1,0) + h\sum_{k=1}^{2^{n-1}} f\left[a + (2k-1)h\right]$$

where $h = (b-a)/2^n$ and $n \geq 1$

$R(1,0) \rightarrow n=1$ $\quad a = -5 \quad b = 0$

$$R(1,0) = \frac{1}{2}R(1-1,0) + h\sum_{k=1}^{2^{n-1}} f(a+(2k-1)h)$$

$$h = \frac{b-a}{2^n} = \frac{0--5}{2^1} = 2.5, \quad R(0,0) = .993755$$

$$R(1,0) = \frac{1}{2}(.993755) + 2.5\sum_{k=1}^{1} f(a+(2k-1)h)$$

$$= \frac{1}{2}(.993755) + 2.5 f(-5+(2\cdot 1-1)2.5)$$

$$= \frac{1}{2}(.993755) + 2.5 f(-2.5)$$

Step 3

Complete the second row and calculate $R(1,1)$

- The calculation of $R(1,1)$ utilizes Richardson's extrapolation, $R(1,1) = f[R(1,0), R(0,0)]$
- The general formula for $R(n,m)$ is

$$R(n,m) = R(n,m-1) + \frac{1}{4^m - 1}[R(n,m-1) - R(n-1,m-1)]$$

$R(1,1) \longrightarrow \underline{n=1}, \underline{m=1} \qquad R(n,m)$

$$R(n,m) = R(n,m-1) + \frac{1}{4^m - 1}\left[R(n,m-1) - R(n-1,m-1)\right]$$

$$R(1,1) = R(1,0) + \frac{1}{4^1 - 1}\left[R(1,0) - R(0,0)\right]$$

Error Analysis

– Cheney and Kincaid (2004)[2] reports the following errors

- The error for the first column is $\mathcal{O}(h^2)$
- The error for the second column is $\mathcal{O}(h^4)$
- The error for the third column is $\mathcal{O}(h^8)$
- and so on

Pseudo-code

Cheney and Kincaid (2004)[3] provide the following pseudo-code

procedure *Romberg*$(f, a, b, n, (r_{ij}))$
real array $(r_{ij})_{0:n \times 0:n}$
integer i, j, k, n

real a, b, h, sum
interface external function f
$h \leftarrow b - a$
$r_{00} \leftarrow (h/2)[f(a) + f(b)]$
for $i = 1$ **to** n **do**
    $h \leftarrow h/2$
    $sum \leftarrow 0$
    for $k = 1$ **to** $2^i - 1$ **step** 2 **do**
        $sum \leftarrow sum + f(a + kh)$
    end for
    $r_{i0} \leftarrow \frac{1}{2} r_{i-1,0} + (sum)h$
    for $j = 1$ **to** i **do**
        $r_{ij} \leftarrow r_{i,j-1} + (r_{i,j-1} - r_{i-1,j-1})/(4^j - 1)$
    end for
end for
end procedure *Romberg*

Romberg Integration Pseudo-code

Python Code for Romberg Integration

```
import math
import numpy as np
def f(z):
    return (math.exp(-0.5*z**2)/((2.0*math.pi)**0.5))
#
a=float(input("Enter the lower limit of the integral: "))
b=float(input("Enter the upper limit of the integral: "))
n=int(input("enter the number of interations (n): "))
#
# initialize matrix r
r=np.zeros(shape=(n+1,n+1))
h=b-a
#find R(0,0)
r[0][0]=(h/2.0)*(f(a)+f(b))
for i in range(1,n+1):
    h=h/2.0
    sum=0.0
    for k in range(1,2**i,2):
        sum=sum+f(a+k*h)
    r[i][0]=0.5*r[i-1][0]+sum*h
    for j in range(1,i+1):
        r[i][j]=r[i][j-1]+(r[i][j-1]-r[i-1][j-1])/(4**j-1)
print(r)
```

Romberg Integration by Hand

Notes

1. Cheny, W., and Kincaid, D., (2004), *Numerical Mathematics and Computer, 5th edition*

2. Cheny, W., and Kincaid, D., (2004), *Numerical Mathematics and Computer, 5th edition*

3. Cheny, W., and Kincaid, D., (2004), *Numerical Mathematics and Computer, 5th edition*