Instructor(s): Dr. Kenneth Duru
First Semester 2019
Mathematical Sciences Institute
Australian National University

# Math3511/6111, Scientific Computing

This Lab book must be submitted by **20th May 5pm**. Late Submissions will incur a 5% penalty per working day. Assignment submissions will close on the **27th May 5pm**. Submissions after this time will be invalid.

# Lab 4: Quadrature

## A. Summary

In this lab you will explore the accuracy of the different quadrature methods. You will explore the Newton-Cotes method using the method of undetermined coefficients. You will also be expected to debug existing code.

The tutors will provide an explanation (and demonstration) of the Newton-Cotes method using the method of undetermined coefficients.

Your task is to implement the Square law and the Trapezoidal law to approximate the following integral

$$\int_{-1}^{1} e^{-3x}\ dx$$

and comment on the accuracy of the two methods.

## B. Labbook – here comes the part which you include or modify (student)

### B1. Left Riemann sum (student) [34 pts]

1) The cell below implements the left Riemann sum for approximating integrals. Before running anything, read through the code and try to understand what it's trying to do. Write a short description on what each procedure is trying to achieve.

Unfortunately, due to the author's wild inexperience with Python, he has made several errors in the following code. Your task is to find each error in the code below, give a brief explaination as to why the error occurs and how to fix the error.

Once you are done fixing the errors in the code, use the square law to approximate the integral

$$\int_{-1}^{1} e^{-3x}\ dx$$

with the partition of the interval ranging from 1 to 100 points. Using the expert knowledge you have gained in the previous labs, along with `scipy.quad` as a reference solution, plot the error of approximation against the size of the partition. Comment on what you observe.

**My Report.**

Your discussion goes here.

```
%matplotlib inline
import numpy as np
import scipy as sp
import math
import scipy.integrate as integrate
import pylab as plt

f = lambda x : numpy.exp(-3*x)

# reference solution of the integral
reference = integrate.quad(lambda x: numpy.exp(-3x),-1,1)[0]


# python function implementing the Riemann sum
def Rect(start = -1, end = 1, f = f, partition_size):
    xpts = np.linspace(start,end,partition_size)
    ypts = f(xpts)
    for i in len(xpts):
        approx = approx + (xpts[i+1]-xpts[i])*f(xpts[i])
    return approx

# Initializing list of error
errvec = ()
for i in range(1,101):
    errvec.append(abs(reference - Rect(i))

# plot of the errors
size = list(range(1,101))
plot(size,errvec,'r-');
```

```
  File "<ipython-input-1-1c120459d176>", line 11
    reference = integrate.quad(lambda x: numpy.exp(-3x),-1,1)[0]
```

```
SyntaxError: invalid syntax
```

### B2. Composite Trapezoidal rule (student) [33 pts]

2) Repeat the accuracy study from point 1 using the Trapezoidal rule. Observe how the error reduces in comparison to the Riemannian sum; comment on wether this is expected.

### My Report.

Your discussion goes here.

### B3. Newton Cotes Method (student) [33 pts]

3) Use the Newton-Cotes formula

$$\int_a^b f(x) = \sum_{i=0}^{4} A_i f(x_i)$$

to estimate the integral

$$\int_{-1}^{1} e^{-3x}$$

with 5 evenly spaced grid points (compare to your reference value).

(Hint: Use the method of undetermined coefficients to solve for the $A_i$, by substituting in $f = 1, x, x^2, x^3, x^4$ and demanding that the result of the integral be exact)

Repeat with 7 and 9 points. Comment on the improvement to your approximations.

### My Report.

Your discussion goes here.