Lab 1

Souradeep Bhattacharya 861105938
EE105 Section: 024

January 18, 2018

Matlab Tutorial

Interaction with command line

```
>> A = [pi;sqrt(2);exp(1)];
>> B = [1;5;7];
>> C = A'*B

C =

29.2406
```

Example script

```
    clear all;
    A = [pi;sqrt(2);exp(1)];
    B = [1;5;7];

    C = 0;
    for i = 1:3
        C = C + A(i)* B(i);
    end
```

The above has C = 29.2406

Function foo

```
function [ y ] = foo( x )
%foo Given an input, x, this function calculates (2/sqrt(pi))*exp(-x^2)
%   Detailed explanation goes here

y = (2/sqrt(pi)) .* exp(-1 .* x .^ 2);

end
```

Function bar

```
function [] = bar( N )
%bar Takes in an N and evenly spaces out region [0, 5] over N + 1 then
```

```
% plots y = foo(x)

x = linspace(0, 5, N + 1);
y = foo(x);

plot(x, y);
title('f(x) vs X');
xlabel('X');
ylabel('f(x)');

end
```

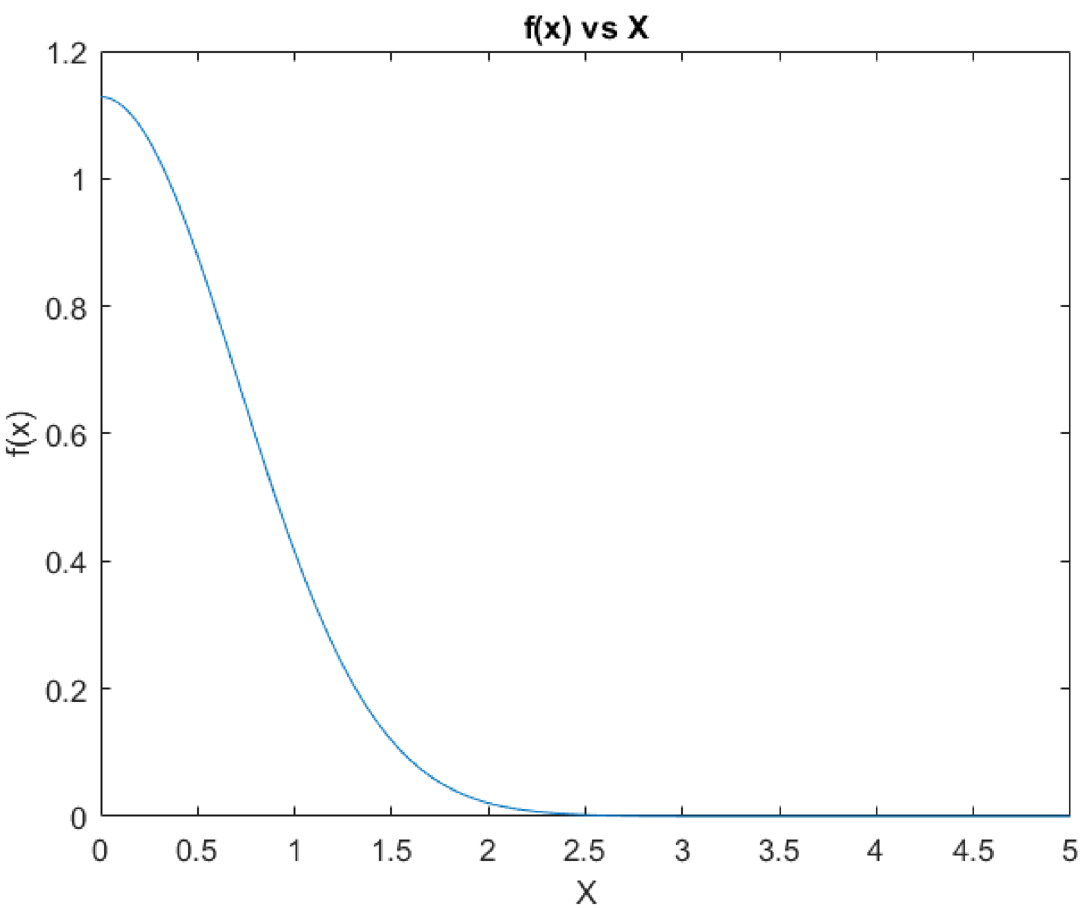

Figure 1: Plot of the given function: $\frac{2}{\sqrt{\pi}}e^{-x^2}$ with N = 200

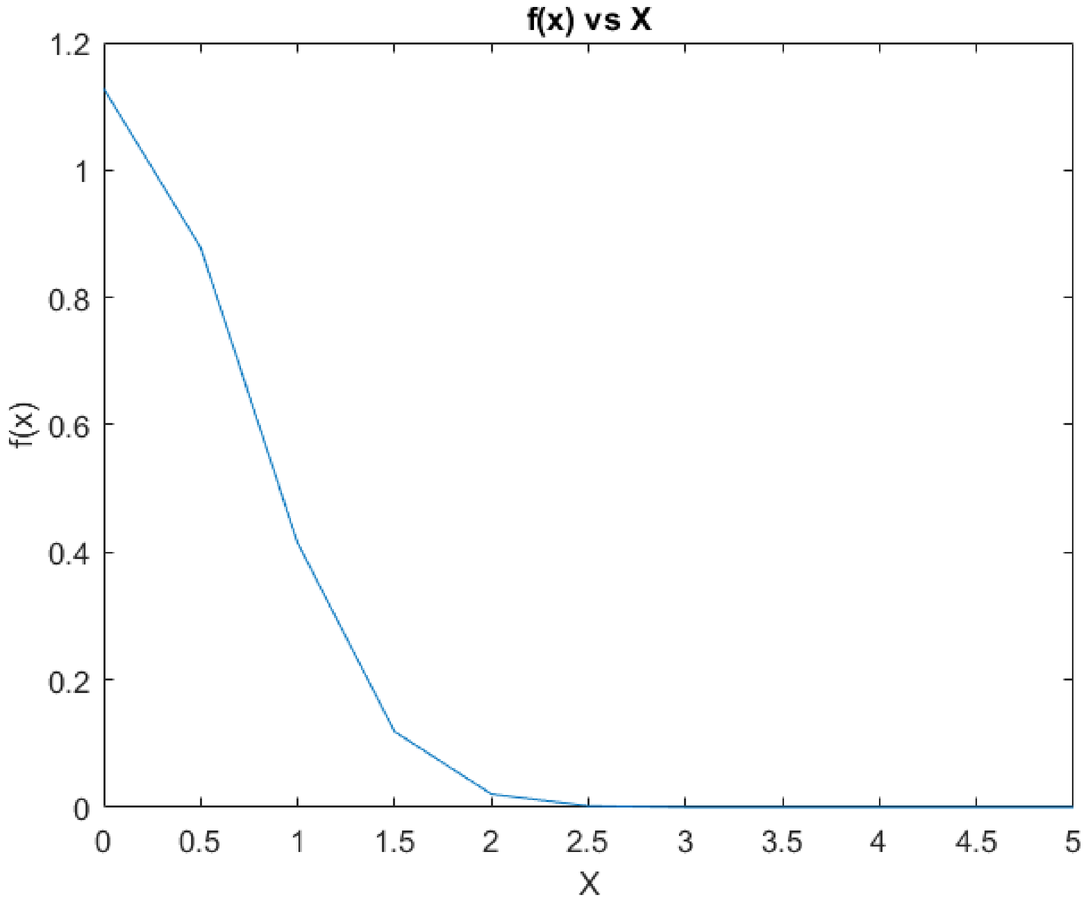

Figure 2: Plot of the given function: $\frac{2}{\sqrt{\pi}}e^{-x^2}$ with N = 10

Integration in Matlab

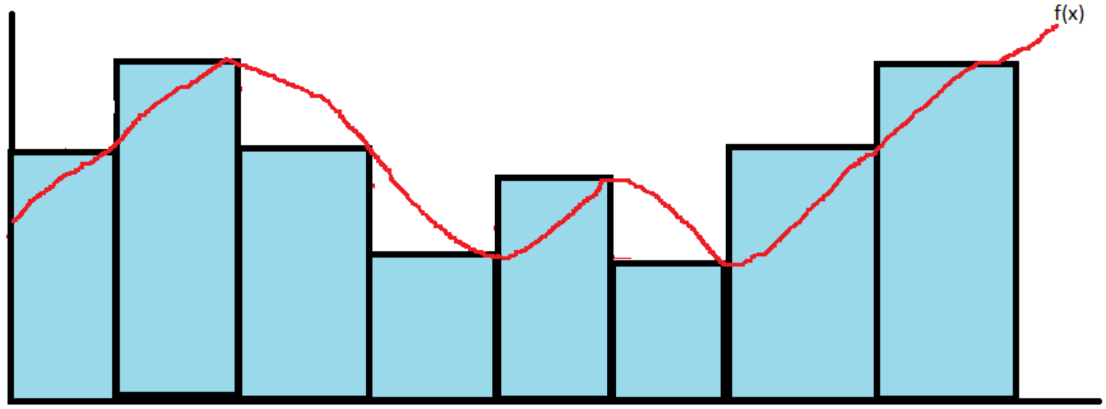

Figure 3: Graphical representation of Right Hand sums (Q2)

Extremely small dx_i reduces the overall error accumulated by either over shoots or undershoots and closely approximates the integral function which uses infinitively small dx_i.

Derivation of Q3

Given that the area under the first line segment can be approximated:

$$f(x_0)dx_0 + 0.5(f(x_1) - f(x_0))dx_0$$

$$f(x_0)dx_0 + \frac{f(x_1)}{2}dx_0 - \frac{f(x_0)}{2}dx_0$$

This means that the next area can be approximated:

$$f(x_1)dx_1 + \frac{f(x_2)}{2}dx_1 - \frac{f(x_1)}{2}dx_1$$

This can continue so forth until the N-th area can be approximated as:

$$f(x_{N-1})dx_{N-1} + \frac{f(x_N)}{2}dx_{N-1} - \frac{f(x_{N-1})}{2}dx_{N-1}$$

Summing these terms up yields the equation Q3

$$Q_3 = f(x_0)\frac{dx_0}{2} + \sum_{i=1}^{N-1} f(x_i)dx_i + f(x_N)\frac{dx_{N-1}}{2}$$

Proof of Q3

Prove $Q_3 = \frac{Q_1+Q_2}{2}$:
Given that:

$$Q_1 = \sum_{i=1}^{N-1} f(x_i)dx_i$$

$$Q_2 = \sum_{i=2}^{N} f(x_i)dx_{i-1}$$

$$Q_3 = f(x_1)\frac{dx_1}{2} + \sum_{i=2}^{N-1} f(x_i)dx_i + f(x_N)\frac{dx_{N-1}}{2}$$

Also all dx_i are equal to each other. Substituting in Q1 and Q2 in to Q3 gives:

$$Q_3 = \frac{1}{2}\left(\sum_{i=1}^{N-1} f(x_i)dx_i + \sum_{i=2}^{N} f(x_i)dx_{i-1}\right)$$

$$Q_3 = \frac{1}{2}\left(f(x_1)dx_1 + \sum_{i=2}^{N-1} f(x_i)dx_i + \sum_{i=2}^{N-1} f(x_i)dx_i + f(x_N)dx_{N-1}\right)$$

$$Q_3 = f(x_1)\frac{dx_1}{2} + \sum_{i=2}^{N-1} f(x_i)dx_i + f(x_N)\frac{dx_{N-1}}{2}$$

Summation Code Q1

```
function [ Q1 ] = Q1_Sum( N )
%Q1_Sum Performs the Q1 Summation technique from the lab manual
%  Conducts a LHS of the function provided by foo.

dx = (5-0)/(N-1);
x = 0:dx:5;

y = foo(x);
A = [ones(N-1,1);0];
Q1 = y*A*dx;
end
```

Summation Code Q3

```
function [ Q3 ] = Q3_Sum( N )
%Q3_Sum Performs the Q3 sum
%  Conducts the Q3 sum, trapizoidal summation.

dx = (5-0)/(N-1);
x = 0:dx:5;

y = foo(x);
A = [0.5;ones(N-2,1);0.5];
Q3 = y*A*dx;
end
```

Plotting Q1

Code

```
% Generates the Q1 Plot and the difference between Q1 and quad.

q = quad(@foo,0,5);
```

```
QA = [];
NA = [];
QD = [];
for N=2:200
    QA = [QA Q1_Sum(N)];
    NA = [NA N];
    QD = [QD (Q1_Sum(N) - q)];
end

plot(NA,QA, NA, QD);
legend('Q1', 'Difference between Q1 and quad');
title('Q1(N) vs N')
xlabel('N');
ylabel('Q1(N)');
```

Figures

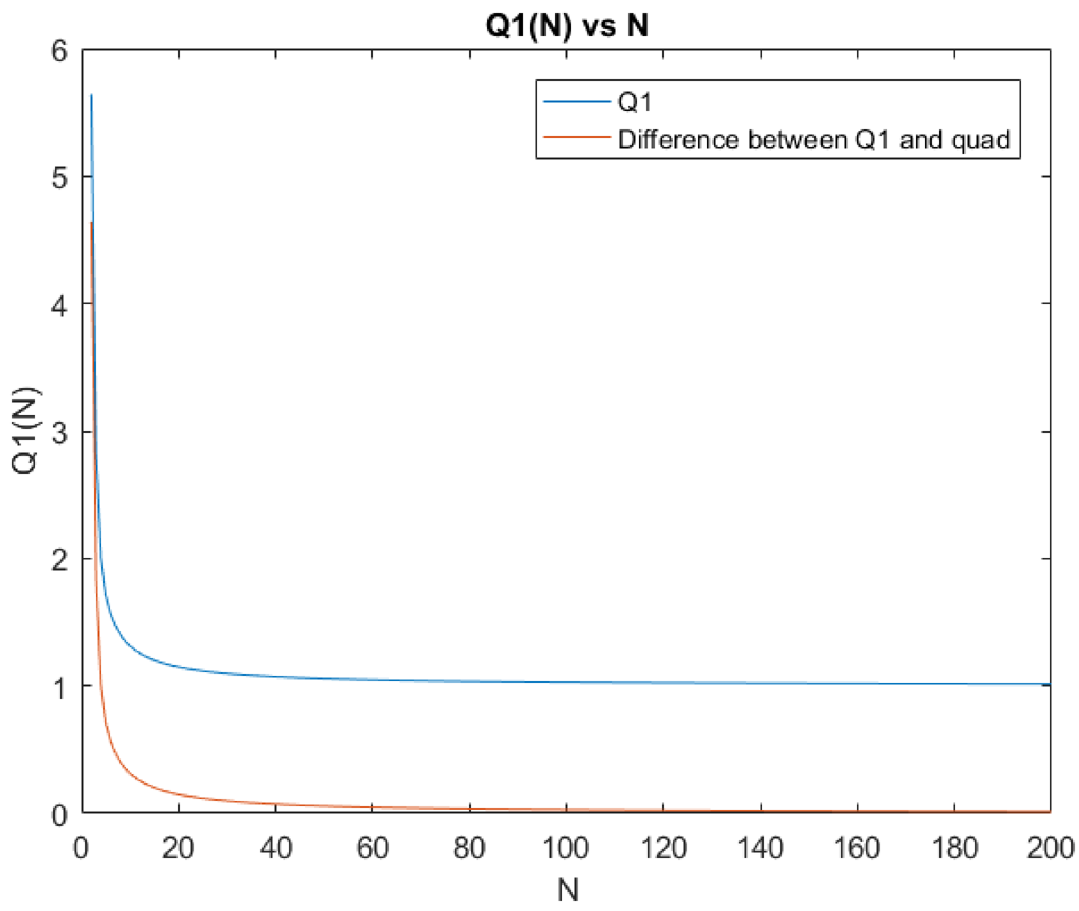

Figure 4: Graphical Representation of Q1(N) versus N and the error between Q1 and quad

Plotting Q3

Code

```matlab
% Generates the Q1 Plot and the difference between Q1 and quad.

q = quad(@foo,0,5);
Q1A = [];
Q3A = [];
NA = [];
Q1D = [];
QD =[];
for N=2:200
   % Calculate the Sums
   Q1 = Q1_Sum(N);
   Q3 = Q3_Sum(N);

   % Difference between Q1 and Q3
   Q1D = [Q1D (Q1-Q3)];
   Q1A = [Q1A Q1];
   Q3A = [Q3A Q3];
   NA = [NA N];
   QD = [QD (Q3 - q)];
end

figure;
plot(NA,Q3A, NA, QD);
legend('Q3', 'Difference between Q3 and quad');
title('Q3(N) vs N')
xlabel('N');
ylabel('Q3(N)');

figure;
plot(NA,Q1A, NA, Q3A, NA, Q1D);
legend('Q1','Q3' ,'Difference between Q1 and Q3');
title('Q(N) vs N')
xlabel('N');
ylabel('Q(N)');
```

Figures

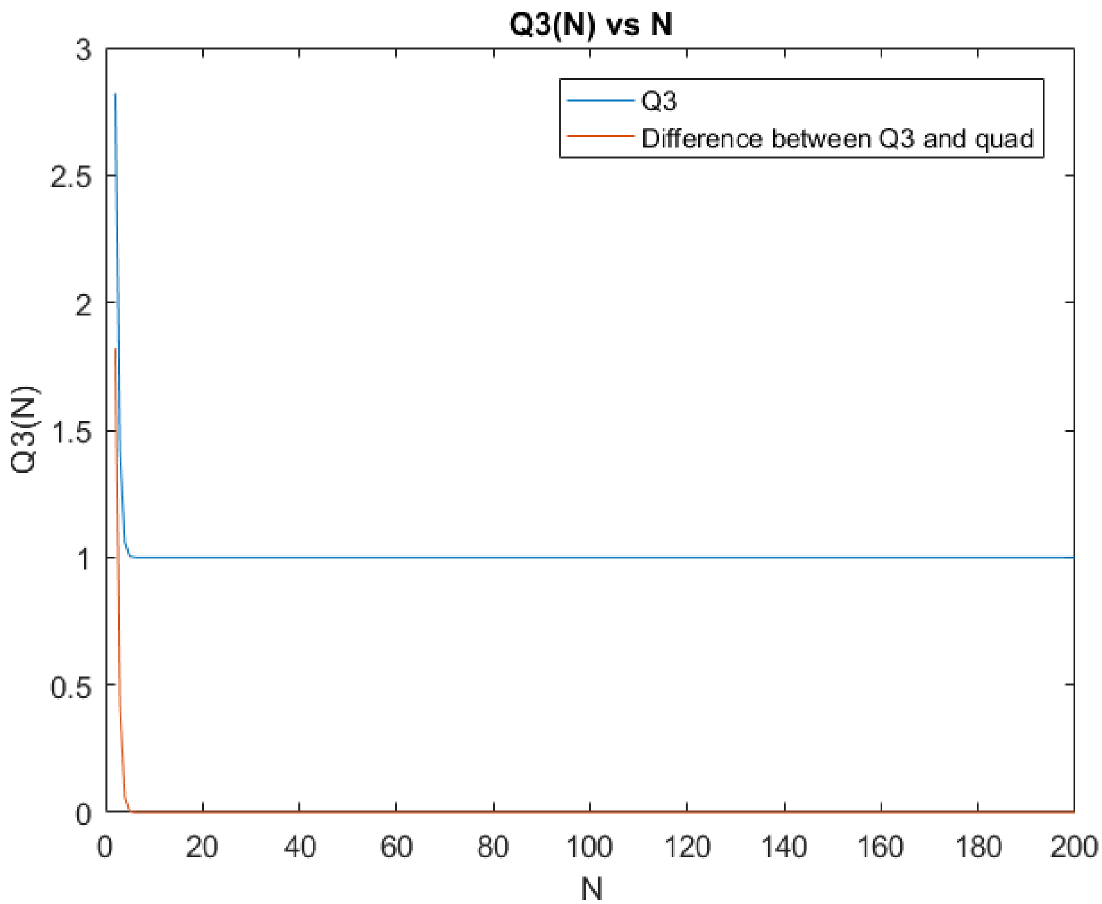

Figure 5: Graphical Representation of Q3(N) versus N and the error between Q3 and quad

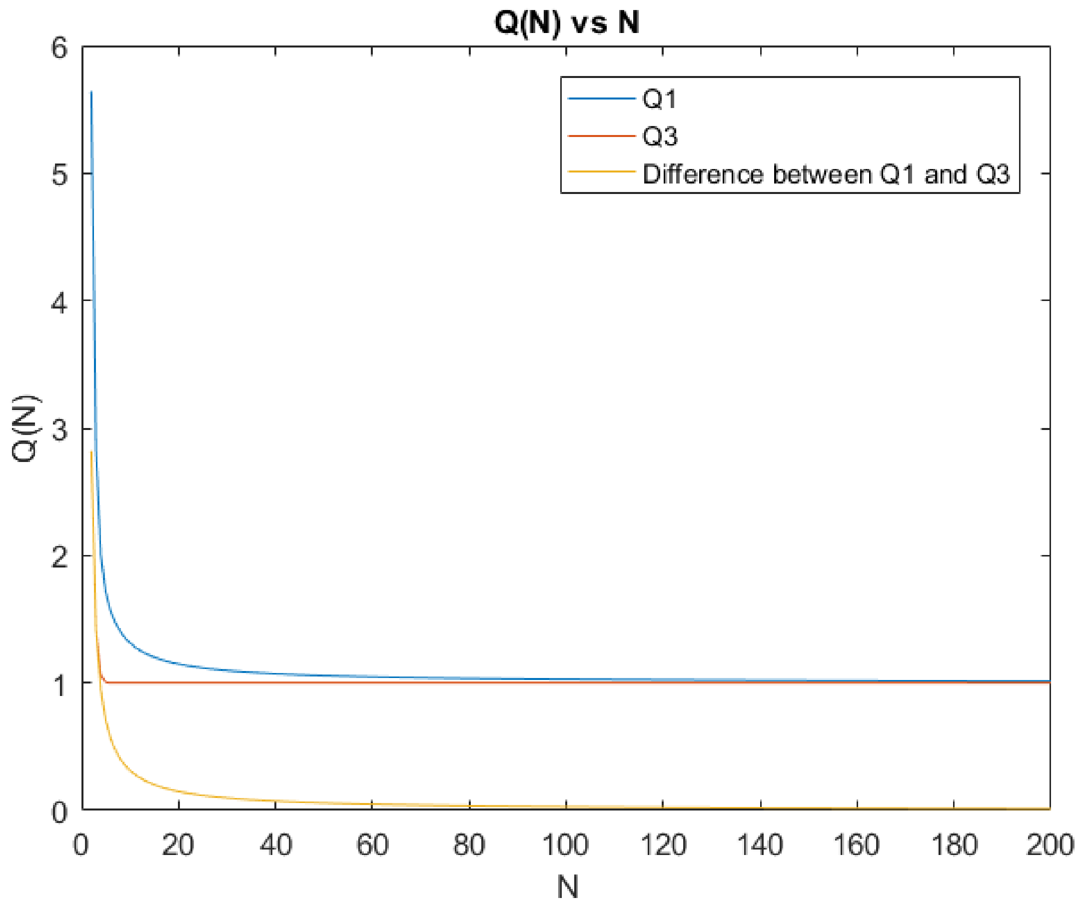

Figure 6: Graphical representation of Q1(N), Q3(N), and the difference between the two.

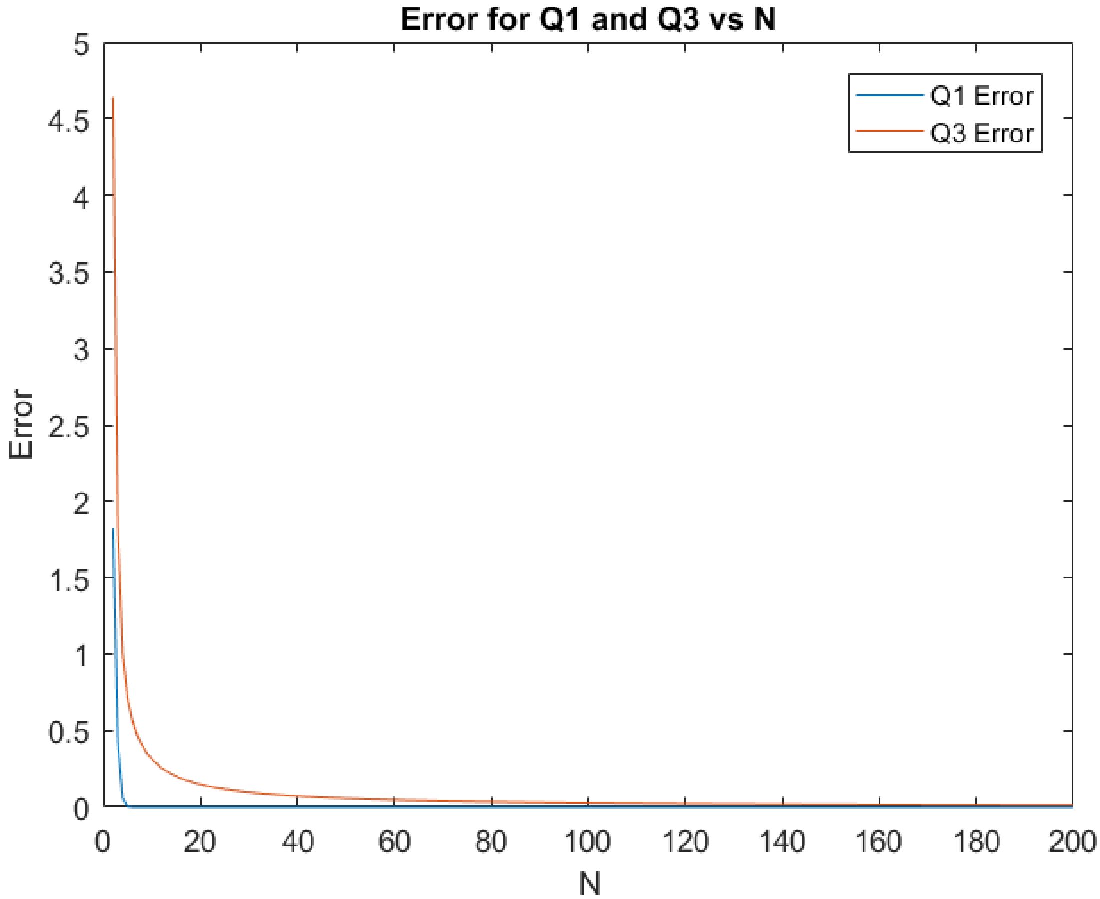

Figure 7: Graphical representation of approximation error between quad and Q1 and quad and Q3.

Based off of my graphical results above the trapezoidal summation, Q3, was the best option. It converged to the same answer provided by quad at a lower N then Q1 did.