MA20222

Numerical Analysis Coursework

Author: Harry Rogers, 209195839

Lecturer: **Clarice Poon**

Date Last Edited: November 12, 2023

1 A1:

Problem 1
show
$$P_1(x) = 2x \tag{1}$$

Solution
$$P_1(x) = \frac{\sin(2\theta)}{\sin(\theta)} = \frac{2\sin(\theta)\cos(\theta)}{\sin(\theta)} = 2\cos(\theta) = 2x \text{ using } \theta := \arccos(x)$$

Problem 2
show
$$P_n(x) = 2xP_{n-1}(x) - P_{n-2}(x) \tag{2}$$

Solution
$$P_n(x) = \frac{\sin((n+1)\theta)}{\sin(\theta)} = \frac{1}{\sin(\theta)}(\sin(n\theta)\cos(\theta) + \cos(n\theta)\sin(\theta)) = \frac{1}{\sin(\theta)}(x\sin(n\theta) + \cos(n\theta)\sin(\theta))$$
$$= \frac{1}{\sin(\theta)}\left[2x\sin(n\theta) - (\sin(n\theta)\cos(-\theta) - \cos(n\theta)\sin(\theta))\right] = \frac{1}{\sin(\theta)}\left[2x\sin(n\theta) - \sin((n-1)\theta)\right]$$
$$= 2xP_{n-1}(x) - P_{n-2}(x)$$

Problem 3
Hence, show that $p_n(x)$ is a polynomial of degree n and that $p_0, ..., p_n$ form a basis for $\mathcal{P}_n$.

Solution
We can use strong induction to prove our polynomial $P_n(x)$ has $\deg(P_n(x)) = n$, and thus show linear independence and spanning properties to show a basis.

Base case: $P_0(x) = 1$, $P_1(x) = 2x$ which have $\deg(P_0(x)) = 0$, $\deg(P_1(x)) = 1$ respectively.

By assuming the statement is true $\forall i \in \{0, 1, ..., k\}$, i.e $\deg(P_i(x)) = i$,
Using 2, we can also write
$$\deg(P_{k+1}(x)) = \deg(2xP_k(x) - P_{k-1}(x))$$

and since $\deg(P_{k-1}(x)) < \deg(P_k(x))$ by our assumption, we yield
$$\deg(P_{k+1}(x)) = \deg(2xP_k(x)) = 1 + \deg(P_k(x)) = k + 1$$

And thus by strong induction we have proven our claim is true for all n.

Furthermore, for each $P_i(x)$ and $P_j(x)$ where $i \neq j$ we have $\deg(P_i(x)) \neq \deg(P_j(x))$
therefore we can conclude linear independence.

For our spanning properties, we find $P_0(x), ..., P_n(x)$ a minimally spanning list of $\mathcal{P}_n$ since
$$\deg\{P_0(x), ..., P(n)\} = \{0, ..., n\}$$

and therefore
$$\mathcal{P}_n = span\{P_0(x), ..., P_n(x)\}$$

Thus we have a basis of $\mathcal{P}_n$ $\square$

2 A2:

Problem 1
Show J_n is non-singular

$$J_n = \begin{bmatrix} P_0(x_1) & P_0(x_2) & \dots & P_0(x_n) \\ P_1(x_1) & P_1(x_2) & \dots & P_1(x_n) \\ P_2(x_1) & P_2(x_2) & \dots & P_2(x_n) \\ \vdots & \vdots & \dots & \vdots \\ P_{n-1}(x_1) & P_{n-1}(x_2) & \dots & P_{n-1}(x_n) \end{bmatrix} \tag{3}$$

Solution
Consider the equation

$$\boldsymbol{c} \in \mathbb{R}^n \;\; s.t \;\; \boldsymbol{c}^T J_n = \boldsymbol{0}$$

By expanding this into its summation we have for

$$\boldsymbol{c} = (c_1, \dots, c_n)$$

$$\boldsymbol{c}^T (J_n)_j = \sum_{i=1}^{n} c_i P_j(x) = 0 \iff c_i = 0 \quad \forall j \leq n$$

therefore we have

$$\boldsymbol{c}^T J_n = 0 \iff \boldsymbol{c} = 0$$

since we have that J_n is injective, we can also claim it has an inverse, thus non-singular. □

Problem 2
Hence, show there exists $w_1, \dots, w_n$ such that

$$\sum_{i=1}^{n} p_k(x_i) w_i = \begin{cases} A, & k = 0, \\ 0, & k = 1, \dots, n-1, \end{cases} \qquad A := \int_{-1}^{1} \sqrt{1 - x^2} dx \tag{4}$$

Solution
Firstly, we set our $\boldsymbol{c} = \boldsymbol{w} = (w_1, \dots, w_n)$ and therefore we have

$$\boldsymbol{c^T} J_n = \begin{bmatrix} \sum_{i=1}^{n} w_i P_0(x_i) \\ \sum_{i=1}^{n} w_i P_1(x_i) \\ \vdots \\ \sum_{i=1}^{n} w_i P_n(x_i) \end{bmatrix}$$

by proving J_n is non-singular, we have also proved that it has full rank n. And therefore, we can say there exists a unique solution to the equation $J_n \boldsymbol{w}^T = \boldsymbol{b}$ for $\boldsymbol{b} = [A, 0, \dots, 0]^T$, and $\boldsymbol{w} = [w_1, w_2, \dots, w_n]^T$.
We prove this using existence and uniqueness:
for uniqueness, Assume $J_n \boldsymbol{w} = \boldsymbol{b}$:

$$J_n^{-1} J_n \boldsymbol{w} = J_n^{-1} \boldsymbol{b} \Longrightarrow \mathcal{I} \boldsymbol{w} = J_n^{-1} b \Longrightarrow \boldsymbol{w} = J_n^{-1} \boldsymbol{b}$$

for existence, assume $\boldsymbol{w} = J_n^{-1} b$:

$$J_n \boldsymbol{w} = J_n J_n^{-1} \boldsymbol{b} = \mathcal{I} \boldsymbol{b} = \boldsymbol{b}$$

Thus we have both existence and uniqueness and we have shown there exists $\boldsymbol{w}$ such that $J_n \boldsymbol{w} = \boldsymbol{b}$
□

3 A3:

Problem 1
by making a substitution, show that:

$$\int_{-1}^{1} P_n(x)P_m(x)\sqrt{1-x^2}dx = 0 \qquad n \neq m \tag{5}$$

Solution
We can start by observing the substitution $x = \cos(u)$, $dx = -\sin(u)du$ changing limits to 0 and π with some algebraic manipulation we can see

$$P_n(x) = \frac{\sin((m+1)u)}{\sin(u)}$$

and therefore our integral can be written as

$$\int_{\pi}^{0} \frac{\sin((m+1)u)}{\sin(u)} * \frac{\sin((n+1)u)}{\sin(u)} * \sqrt{1-(\cos(u))^2} * -\sin(u)du$$

by writing $\sqrt{1-(cos(u))^2} = \sin(u)$, cancelling the $\sin(u)$'s and changing the bounds by using the negative sign we yield

$$\int_{0}^{\pi} \sin((m+1)u) * \sin((n+1)u)du$$

we need a way to split up our $\sin((m+1)u)\sin((n+1)u)$.
Consider $\cos((a-b)x) - \cos((a+b)x)$ and expand:

$$\cos((a-b)x) - \cos((a+b)x) = \cos(ax)\cos(bx) - \sin(ax)\sin(bx) - [\cos(ax)\cos(bx) - \sin(ax)\sin(bx)]$$

simplifying like terms leads us to

$$2\sin(ax)\sin(bx) = \cos((a-b)x) - \cos((a+b)x)$$

And therefore, by substituting the identity and $a = m+1, \;\; b = n+1$ we can find our integral to be

$$\frac{1}{2}\int_{0}^{\pi} \cos((m-n)u) - \cos((m+n+2)u)du$$

integrating gives us

$$\frac{1}{2}\left[\frac{1}{m-n}\sin((m-n)x) - \frac{1}{m+n+2}\sin((m+n+2)x)\Big|_0^{\pi}\right]$$

for $x = 0$ we evaluate to 0. Then for π

$$\frac{1}{2}\left[\frac{1}{m-n}\sin((m-n)\pi) - \frac{1}{m+n+2}\sin((m+n+2)\pi)\right] = 0$$

This is because any multiple $k\pi$ in sin is 0, i.e $\sin(k\pi) = 0 \quad \forall k \in \mathbb{Z}$ $\square$

4 A4:

Problem 1
By using the basis $\{p_0, \ldots, p_n\}$ for $\mathcal{P}_n$ and A3, prove that

$$\int_{-1}^{1} p(x)\sqrt{1-x^2}dx = \sum_{i=1}^{n} p(x_i)\, w_i, \quad \forall p \in \mathcal{P}_n$$

for the x_i and w_i defined above.

Solution
we start by noticing we can write $p(x) = \sum_{i=0}^{n} P_i(x)\lambda_i$ since we already have a basis of $\mathcal{P}_n$ in terms of $\{p_0, \ldots, p_n\}$, and can therefore write our integral as:

$$\int_{-1}^{1} \sum_{i=0}^{n} \lambda_i P_i(x)\sqrt{1-x^2}dx$$

We can interchange the summation and the integral due to the sum being absolutely convergent:

$$\sum_{i=0}^{n} \lambda_i \int_{-1}^{1} P_i(x)\sqrt{1-x^2}dx = \lambda_0 \int_{-1}^{1} P_0(x)\sqrt{1-x^2}dx$$

since from 5 we find all values of $i \neq 0$ cancel.
Furthermore, using our basis fact on the right hand side, and 4 for the last step:

$$\sum_{i=1}^{n} p(x_i)w_i = \sum_{i=1}^{n} w_i \left(\sum_{k=0}^{n} p_k(x_i)\lambda_k\right) = \sum_{k=0}^{n} w_i \left(\sum_{i=1}^{n} p_k(x_i)\lambda_k\right) = \lambda_0 \sum_{i=1}^{n} p_0(x_i)w_i$$

using 4 to equate these statements, we have the following:

$$\lambda_0 \int_{-1}^{1} P_0(x)\sqrt{1-x^2}dx = \lambda_0 \sum_{i=1}^{n} p_0(x_i)w_i \iff \int_{-1}^{1} p(x)\sqrt{1-x^2}dx = \sum_{i=1}^{n} p(x_i)w_i$$

for some $p(x) \in \mathcal{P}_n$ $\square$

Problem 2
Now show that the relationship also holds for $p \in \mathcal{P}_{2n-1}$ to show the quadrature rule has degree of precision at least $2n-1$ and hence is a Gaussian quadrature rule.

Solution
We start by writing $p(x) \in \mathcal{P}_{2n-1}$ and deducing there must be a polynomial with $q(x),\ r(x) \in \mathcal{P}_{n-1}$ such that

$$p(x) = P_n(x)q(x) + r(x)$$

splitting our integral we have

$$\int_{-1}^{1} \left[p_n(x)q(x) + r(x)\right]\sqrt{1-x^2}dx = \int_{-1}^{1} p_n(x)q(x)\sqrt{1-x^2}dx + \int_{-1}^{1} r(x)\sqrt{1-x^2}dx$$

since $\deg(P_n(x)) > \deg(q(x))$ we find our first integral resolves to zero, and using what we have shown above, we find:

$$\int_{-1}^{1} r(x)\sqrt{1-x^2}dx = \sum_{i=1}^{n} p(x_i)w_i$$

and therefore we have shown we have a degree of precision of at least $2n-1$ and hence have a Gaussian quadrature rule. $\square$

5 A5:

Problem 1
Let $q_0(x) := 1$ and $q_n(x) := 2^n \det(xI - A_n)$ for $n = 1, 2, \ldots$, where A_n is the $n \times n$ matrix

$$A_n := \begin{bmatrix} 0 & 1/2 & 0 & \cdots & & & \\ 1/2 & 0 & 1/2 & \ddots & & & \\ 0 & 1/2 & 0 & 1/2 & & & \\ \vdots & \ddots & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & & 1/2 \\ & & & & 1/2 & 0 \end{bmatrix}$$

(all entries zero, apart from the two main off-diagonals). Show that $q_n = P_n$ for $n = 1, 2, \ldots$

Solution
Our proof of this statement will use the fact $2^n \det(x\mathcal{I} - A_n) = \det(2(x\mathcal{I} - A_n))$ and by using cofactors to split up our determinant. We start by showing

$$q_n(x) := 2^n \det(x\mathcal{I} - A_n) = \det(2(x\mathcal{I} - A_n)) := \begin{vmatrix} 2x & -1 & 0 & \cdots & & \\ -1 & 2x & -1 & \ddots & & \\ 0 & -1 & 2x & -1 & & \\ \vdots & \ddots & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & & -1 \\ & & & & -1 & 2x \end{vmatrix}$$

furthermore, we can split this into cofactors:

$$2xq_{n-1}(x) + \begin{vmatrix} -1 & -1 & 0 & \cdots & & \\ 0 & 2x & -1 & \ddots & & \\ 0 & -1 & 2x & -1 & & \\ \vdots & \ddots & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & & -1 \\ & & & & -1 & 2x \end{vmatrix} = 2xq_{n-1}(x) - q_{n-2}(x) - \begin{vmatrix} 0 & 0 & 0 & \cdots & & \\ 0 & 2x & -1 & \ddots & & \\ 0 & -1 & 2x & -1 & & \\ \vdots & \ddots & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & & -1 \\ & & & & -1 & 2x \end{vmatrix}$$

$$= 2xq_{n-1}(x) - q_{n-2}(x)$$

Therefore we can see that we have the relation $q_n(x) = 2xq_{n-1}(x) - q_{n-2}(x)$ and since both have the same initial values $q_0(x) := 1$, $P_0(x) := 1$ and $q_1(x) = 2\det(x) = 2x$, $P_1(x) = 2x$. we can conclude that we have equality, i.e $q_n(x) = P_n(x)$ □

Problem 2
Hence, show that eigenvalues of A_n equal the quadrature nodes x_i .

Solution
we have our eigenvalues given by $\det(A_n - \mathcal{I}\lambda) = 0$ which we can rearrange to be
$(-1)^n \det(\mathcal{I}\lambda - A_n) = 0$ which we divide through by $(-1)^n$ and yield

$$q_n(\lambda) = P_n(\lambda) = 0 \iff \lambda_i = x_i \quad \forall x_i \in (x_1, x_2, \ldots, x_n)$$

□

6 A6:

Problem 1
Denote the i th column of J_n (from A2) by

$$\boldsymbol{v}^i := \left[p_0\left(x_i\right), p_1\left(x_i\right), \ldots, p_{n-1}\left(x_i\right)\right]^\top, \quad i = 1, \ldots, n$$

Show that $\boldsymbol{v}^i$ is an eigenvector of A_n corresponding to the eigenvalue x_i.

Solution
We can solve this by noticing if $\boldsymbol{v}^i$ is an eigenvector then we have $|A_n - \lambda\mathcal{I}|\boldsymbol{v}^i = 0$ for all i thus by working with $|A_n - \lambda\mathcal{I}|\boldsymbol{v}^i$

$$|A_n - \lambda\mathcal{I}|\boldsymbol{v}^i = 0 \quad \forall i \tag{6}$$

thus we can expand 6 for some λ_i to gain the set of equations

$$\begin{aligned} -\lambda_i P_0(x_i) + \frac{1}{2}P_1(x_i) \\ \frac{1}{2}P_0(x_i) - \lambda_i P_1(x_i) + \frac{1}{2}P_2(x_i) \\ \vdots \qquad \vdots \qquad \vdots \\ \frac{1}{2}P_{n-3}(x_i) - \lambda_i P_{n-2}(x_i) + \frac{1}{2}P_{n-1}(x_i) \end{aligned}$$

If in 2 we set our $x = x_i = \lambda_i$ then we have:

$$P_{n-1}(x_i) = 2\lambda_i P_{n-2}(x_i) - P_{n-3}(x_i) \implies \frac{1}{2}P_{n-3}(x_i) - \lambda_i P_{n-2}(x_i) + \frac{1}{2}P_{n-1}(x_i) = 0$$

Thus we have shown for $x = x_i = \lambda_i$ we yield 0 and therefore $\boldsymbol{v_i}$ $\forall i \leq n$ are eigenvectors of x_i respectively. □

Problem 2
Using the fact that eigenvectors of a symmetric matrix are orthogonal, show that the quadrature weight satisfy

$$w_i = A\frac{1}{\|v^i\|^2}\left(v_1^i\right)^2, \quad A := \int_{-1}^{1}\sqrt{1-x^2}dx = \frac{1}{2}\pi$$

where v_1^i is the first component of the i th eigenvector v^i.

Solution
We start by noting for orthogonal vectors we have $(v^i)(v^i)^T = ||v^i||^2$, and by using 4 we can generalise $J_n w = A\boldsymbol{e_1}$ where $\boldsymbol{w} = [w_1, w_2, \ldots, w_n]^T$ and $\boldsymbol{e_1} = [1, 0, 0, \ldots, 0]^T$. Furthermore, $v_1^i = P_0(x_i) = 1$.
thus multiplying through by $(v^i)^T$ we have $(v^i)^T J_n\boldsymbol{w} = (v^i)^T A\boldsymbol{e_1}$ and generalising for the i^{th} column (noticing $(J_n)_i = v^i$ given by the definition of v^i) our equation becomes:

$$(v^i)^T(v^i)w_i = (v^i)^T A\boldsymbol{e_1} \implies ||v^i||^2 w_i = A(v^i)^T\boldsymbol{e_1}$$

then multiplying through on the right hand side we find that $(v^i)^T\boldsymbol{e_1} = p_0(x_i) = 1$ then rearranging yields

$$w_i = \frac{A}{||v^i||^2} = \frac{A}{||v^i||^2}(v_1^i)^2$$

as required. □

7 B1:

As convention, we define anything below 10^{-16} as floating point error.

Problem 1
Write a MATLAB code to compute the quadrature nodes $x_1, \ldots, x_n$, and weights $w_1, \ldots, w_n$ for the quadrature rule developed in §A

Solution

```
function [x,w]=getquad(n)
% Return a vector of quadrature nodes x and weights w, of dimension n.
%construct A_n
A_n = diag(0.5*ones(1,n-1),1) + diag(0.5*ones(1,n-1),-1);
[V,x] = eig(A_n,'vector');

% magnitutude of w1i in MATLAB will always be 1 due to MATLAB unit normalising
% the eigenvectors

%V1i
V1 = V(1,:);
w = 0.5*pi.*V1.^2;
end
```

This will result in the quadrature nodes and weights for some n by the command [x,w] = getquad(n)

8 B2:

Problem 1
Write a routine to evaluate the quadrature for a given function g.
It should run with the call myquad(@(x) x,x,w) to evaluate $\int_{-1}^{1} x\sqrt{1-x^2}dx$

Solution

```
function out=myquad(g,x,w)
assert (length(x)==length(w)) % error check to ensure x and w are equal in dimension

n = length(x);
% Evaluate sum_{i=1,...,n} w_i g(x_i)

sum = 0; % Initialise the sum as 0
% Iterate over i to find our total sum of w_i * g(x_i)
for i=1:n
    sum = sum + w(i)*g(x(i));
end
out=sum;
% Returns value
end
```

by entering myquad(@(x) x,x,w) to evaluate $\int_{-1}^{1} x\sqrt{1-x^2}dx$ for $n = 4$ we have

```
>> myquad(@(x) x,x,w)

ans =

 -1.1102e-16
```

which is 0 with floating point precision, typing

```
>> integral(@(x) x.*sqrt(1-x.^2),-1,1)

ans =

    -2.42861286636753e-17
```

to evaluate using built in MATLAB functions also gives 0 to floating point precision.

Problem 2
Verify the degree of precision is $2n-1$ for $n=10$

Solution
Below we have code that verifies that the DOP is $2n-1$

```
n=10;
[x,w] = getquad(10);
accuracy=[];
k=1;
for i=1:2*n
    quadrature(i) = myquad(@(x) x^i,x,w);
    % Fetches quadrature rule developed in B2
    int(i) = integral(@(x) x.^i .* sqrt(1-x.^2),-1,1);
    % Using MATLAB to evaluate the integral
    accuracy(i) = abs(quadrature(i) - int(i));
    if accuracy(i) > (10e-16) % Defining a tolerance
        DOP(k) = i;
        k=k+1; % Iterates for the next array value of DOP
    end
end
DOP = min(DOP)-1
```

running this code gives us:

```
>> B2_DOP
DOP =
    19
```

Considering the polynomials in a table (omitting the previous results due to them being 0) we have:

polynomial $g(x)$	quadrature $Q(g(x))$	integral $I(g(x))$	$E = \|Q(g(x)) - I(g(x))\|$
x^{16}	0.0342748832293197	0.0342748832293198	0
x^{17}	-2.77555756156289e-17	3.46944695195361e-18	0
x^{18}	0.0291336507449217	0.0291336507449218	0
x^{19}	-2.77555756156289e-17	6.93889390390723e-18	0
x^{20}	0.0251593821606829	0.0251608801887961	1.49802811326427e-06
x^{21}	-3.12250225675825e-17	2.60208521396521e-18	0
x^{22}	0.0220082800246307	0.0220157701651966	7.49014056593278e-06

We can conclude that the DOP is $2n-1$ for $n=10$

9 B3:

Problem 1
use built in MATLAB functions to evaluate $\int_{-1}^{1} g(x)\sqrt{1-x^2}$ for $g_1(x) = \exp(x)$ and $g_2(x) = x\sin(x)$

Solution

```
function [Int1,Int2] = B3()

% Defining a function handle for g
f1 = @(x) exp(x).*sqrt(1-x.^2);
f2 = @(x) x.*sin(x).*sqrt(1-x.^2);

%using built in functions to evaluate the integrals
Int1 = integral(f1,-1,1);
Int2 = integral(f2,-1,1);

end
```

we can reference our integrals by calling [Int1,Int2] = B3()

10 B4:

Problem 1
Using the quadrature method developed in B2, evaluate

$$\int_{-1}^{1} g_i(x)\sqrt{1-x^2}dx, \qquad \text{for } i = 1, 2.$$

Using semilogy, give a plot of the error against n for $n = 1, 2, \ldots, 20$.

Solution

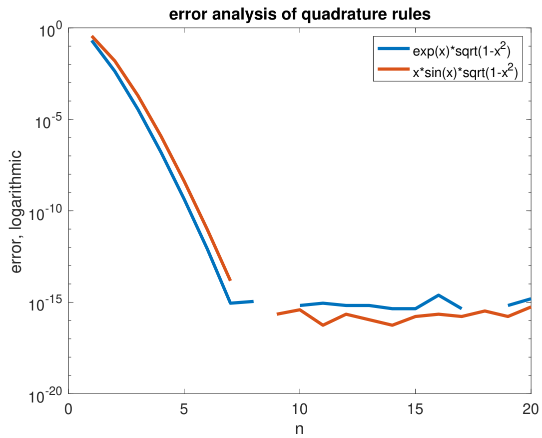
```
% Define our list n and call the integrals from B3
n=1:20;
[Int1,Int2]=B3();

% Run a for loop i = 1 to n to find the error at each quadrature rule
for i = 1:max(n)
    [x,w] = getquad(i);
    myquad1 = myquad(@(x)exp(x),x,w);
    myquad2 = myquad(@(x)x*sin(x),x,w);
    error1(i) = abs(Int1-myquad1);
    error2(i) = abs(Int2-myquad2);
end

% Using semilogy to get a logarithmic graph of our errors for different points n for g_1
semilogy(n,error1,'-','lineWidth',3);
title('error analysis of quadrature rules');
ylabel('error, logarithmic');
xlabel('n');
set(gca,'FontSize',14);
```

```
% Adding our errors for g_2
hold on
semilogy(n,error2,'-','lineWidth',3);
hold off
```

Yields the following graph:

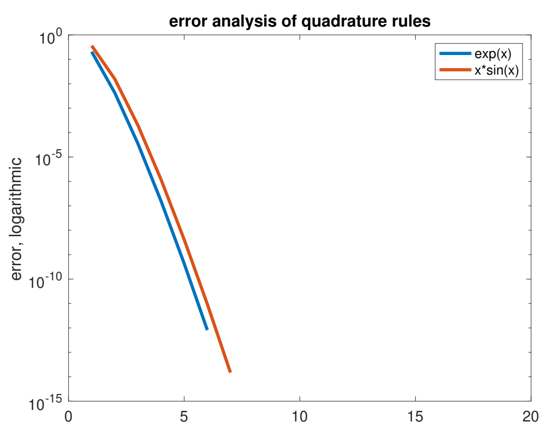


We note for the discontinuities in the graph that our error is zero in MATLAB. Furthermore, if we add a conditional statement to the for loop to omit errors below 10^{-16}:

```
    for i = 1:max(n)
    [x,w] = getquad(i);
    myquad1 = myquad(@(x) exp(x),x,w);
    myquad2 = myquad(@(x) x*sin(x),x,w);

    error1(i) = abs(Int1-myquad1);
    error2(i) = abs(Int2-myquad2);
    if error1(i) < 10e-16
        error1(i) = 0;
    end
    if error2(i) < 10e-16
        error2(i) = 0;
    end
end
```

This gives us the following graph:



We do this to allow for floating point errors within MATLAB.