Table of Contents

Q1

```
clear;

n = 1000;

x = haltonseq(n,2);
y = x(:,2);
x = x(:,1);

sum1 = 0;

for i = 1 : n
    xi = x(i,1);
    yi = y(i,1);

    if xi^2 + yi^2 <= 1
        sum1 = sum1 + 1;
    else sum1 = sum1 + 0;
    end
end

sum1 = (1/n)*sum1;
sum1 = 4*sum1;
```

Q2

```
h = 1/n;

x = 0:h:1;
y = 0:h:1;

x = x';
y = y';

w = 4*ones(length(x));
w(1,:) = (1/2)*w(1,:);
w(end,:) = (1/2)*w(end,:);
w(:,1) = (1/2)*w(:,1);
w(:,end) = (1/2)*w(:,end);
w=w*(h/2)^2;
```

```
fx = [];


for i = 1 : n+1
    for j = 1 : n+1

    xi = x(i,1);
    yi = y(j,1);

    if xi^2 + yi^2 <= 1
        fx(i,j) = 1;
    else
        fx(i,j) = 0;
    end
    end
end


sum2 = (1/n)*sum(sum((fx'*w)));
sum2 = 4*sum2;
```

Q3

```
x = haltonseq(n,1);

sum3 = 0;

for i = 1 : n

    xi = x(i,1);
    sum3 = sum3 + sqrt( 1 - xi^2 );

end

sum3 = (1/n)*sum3;
sum3 = 4*sum3;
```

Q4

```
h = 1/n;

x = 0:h:1;
x = x';

w = 2*ones(length(x),1);
w(1) = 1;
w(end) = 1;
w=w*(h/2);

fx = sqrt( 1 - x.^2 );

sum4 = fx'*w;
sum4 = 4*sum4;
```

Q5

```
MSE1n = [];
MSE2n = [];
MSE3n = [];

for n = [100 1000 10000]

    mse1 = [];
    mse2 = [];
    mse3 = [];

    for r = 1:200


        x1 = haltonseq(n,1);
        x2 = rand([n,1]);
        h = 1/n;
        x3 = 0:h:1;
        x3 = x3';

        sum51 = 0;
        sum52 = 0;

        for i = 1 : n

            x1i = x1(i,1);
            x2i = x2(i,1);
            sum51 = sum51 + sqrt( 1 - x1i^2 );
            sum52 = sum52 + sqrt( 1 - x2i^2 );

        end

        % Quasi-MC integration
        sum51 = 4*(1/n)*sum51;
        % Pseudo-MC integration
        sum52 = 4*(1/n)*sum52;

        w = 2*ones(length(x3),1);
        w(1) = 1;
        w(end) = 1;
        w=w*(h/2);

        fx = sqrt( 1 - x3.^2 );

        sum53 = fx'*w;
        % Newton-Coates method
        sum53 = 4*sum53;

        bias1 = ( sum51 - pi )^2;
        bias2 = ( sum52 - pi )^2;
        bias3 = ( sum53 - pi )^2;
```

```
        mse1 = [ mse1 ; bias1 ];
        mse2 = [ mse2 ; bias2 ];
        mse3 = [ mse3 ; bias3 ];

    end

    MSE1 = mean(mse1);
    MSE2 = mean(mse2);
    MSE3 = mean(mse3);

    MSE1n = [ MSE1n MSE1 ];
    MSE2n = [ MSE2n MSE2 ];
    MSE3n = [ MSE3n MSE3 ];

end

MSE = [ MSE1n ; MSE2n ; MSE3n ];
MSE

% Overall, as n gets bigger, mean squared error is getting smaller.
% Compared to pseudo-MC integration results (second row), you can
 check
% quasi-MC integration and Newton-Coates method do better. Among them,
% Newton-Coates method is the best.
```

```
MSE =

   0.001588178044260   0.000023062446244   0.000000430027979
   0.007208317783551   0.000718405425048   0.000083071984709
   0.000001382086642   0.000000001382849   0.000000000001383
```

Published with MATLAB® R2018b