Table of Contents

..... 1
Question 1a 1
Question 1b 1
Question 2a 2
Question 2b 3
Graph for 2 3


```
%%HW#3 Name: Michael Bautista, email:mbautis000@citymail.cuny.edu
```

```
Error using dbstatus
Error: File: /MATLAB Drive/Michael_Bautista_HW3.m Line: 25 Column: 1
Function definitions in a script must appear at the end of the file.
Move all statements after the "hw4_fn_h" function definition to before
 the first local function definition.
```

Question 1a

```
close all; clear all; clc

y = hw4_fn_h(rand(100000,100)) ;

y_mean = mean(y)
y_std = std(y);

N = size(y,1);                                          % Number of
 'Experiments' In Data Set
y_error = std(y)/sqrt(N);

CI95 = tinv([0.025 0.975], N-1);            % Calculate 95%
 Probability Intervals Of t-Distribution
yCI95 = bsxfun(@times, y_error, CI95(:))          % Calculate 95%
 Confidence Intervals Of All Experiments At Each Value Of _xs

interval = y_mean + yCI95

function h=hw4_fn_h(x)
% x should be a Nx100 matrix
if size(x,2) ~= 100, error('wrong size'), end
h=abs( sin( 2*pi*x(:,1).*sum(x,2))).*((cos( 2*pi*x(:,2).*
 sum(x.^2,2))).^2);
end
```

Question 1b

```
close all; clear all; clc

y = hw4_fn_h(rand(100000,100)) *1.05.^100 ;
```

```
y_mean = mean(y)
y_std = std(y);

N = size(y,1);                                          % Number of
 'Experiments' In Data Set
y_error = std(y)/sqrt(N);

CI95 = tinv([0.025 0.975], N-1);            % Calculate 95%
 Probability Intervals Of t-Distribution
yCI95 = bsxfun(@times, y_error, CI95(:))          % Calculate 95%
 Confidence Intervals Of All Experiments At Each Value Of _xs

interval = y_mean + yCI95

%The answer for 1b is much larger than 1a because we are multiplying
 it
%by 1.05^100.

function h=hw4_fn_h(x)
% x should be a Nx100 matrix
if size(x,2) ~= 100, error('wrong size'), end
h=abs( sin( 2*pi*x(:,1).*sum(x,2))).*((cos( 2*pi*x(:,2).*
 sum(x.^2,2))).^2);
end
```

Question 2a

```
close all; clear all; clc

samples = 10^6;
r = ones(samples,1); %Matrix of temporary values to be replaced

for i = 1:samples
r(i,1) = question2_fn(rand,rand,1);
end

r_mean = mean(r);
r_std = std(r);

N = size(r,1);
r_error = std(r)/sqrt(N);

CI95 = tinv([0.025 0.975], N-1);            % Calculate 95%
 Probability Intervals Of t-Distribution
rCI95 = bsxfun(@times, r_error, CI95(:))          % Calculate 95%
 Confidence Intervals Of All Experiments At Each Value Of _xs

interval = r_mean + rCI95

function h=question2_fn(x,y,z)
    if sqrt(x.^2 + y.^2) >= 0 & sqrt(x.^2 + y.^2) <= 0.5
        h=z.*(cos(3.*x.*y)).^2 .*exp(-z.*(x.^2+y.^2));
```

```
    end

    if sqrt(x.^2 + y.^2) > 0.5 & sqrt(x.^2 + y.^2) <= 1
        h=z.*exp(x.*y).*exp(-z.*(x.^2+y.^2));
    end

    if sqrt(x.^2 + y.^2) > 1
        h = 0;
    end
end
```

Question 2b

```
close all; clear all; clc

samples = 10^6;
r = ones(samples,1); %Matrix of temporary values to be replaced

for i = 1:samples
r(i,1) = question2_fn(rand,rand,50);
end

r_mean = mean(r);
r_std = std(r);

N = size(r,1);
r_error = std(r)/sqrt(N);

CI95 = tinv([0.025 0.975], N-1);            % Calculate 95%
 Probability Intervals Of t-Distribution
rCI95 = bsxfun(@times, r_error, CI95(:))          % Calculate 95%
 Confidence Intervals Of All Experiments At Each Value Of _xs

interval = r_mean + rCI95

function h=question2_fn(x,y,z)
    if sqrt(x.^2 + y.^2) >= 0 & sqrt(x.^2 + y.^2) <= 0.5
        h=z.*(cos(3.*x.*y)).^2 .*exp(-z.*(x.^2+y.^2));
    end

    if sqrt(x.^2 + y.^2) > 0.5 & sqrt(x.^2 + y.^2) <= 1
        h=z.*exp(x.*y).*exp(-z.*(x.^2+y.^2));
    end

    if sqrt(x.^2 + y.^2) > 1
        h = 0;
    end
end
```

Graph for 2

```
clear all; close all; clc
h1 = @(x,y,z) z.*(cos(3*x.*y))^2.*exp(-z.*(x^2 + y^2)); % inner
```

```
h2 = @(x,y,z) z.*exp(x.*y).*exp(-z.*(x^2 + y^2)); % outer

p = 36; % meshgrid resolution

figure

subplot(2,1,1)

[R1, T] = meshgrid(linspace(0,0.5,p),linspace(0,2*pi,p));
X1 = R1.*cos(T);
Y1 = R1.*sin(T);
Z11 = arrayfun(h1,X1,Y1,ones(p));
surf(X1,Y1,Z11)

hold on

R2 = linspace(0.5,1,p);
X2 = R2.*cos(T);
Y2 = R2.*sin(T);
Z21 = arrayfun(h2,X2,Y2,ones(p));
surf(X2,Y2,Z21)

subplot(2,1,2)

Z12 = arrayfun(h2,X1,Y1,50*ones(p));
surf(X1,Y1,Z12)

hold on

Z22 = arrayfun(h2,X2,Y2,50*ones(p));
surf(X2,Y2,Z22)

axis([-1 1 -1 1 0 50])
```

Published with MATLAB® R2019a