EGR 103L – Fall 2016

# Laboratory 4 - Loops and Plots

CEMAL YAGCIOGLU (cy111)
Lab Section EGR 103L9-05, WEDNESDAY 11.45 PM - 2.35 AM
2 OCTOBER, 2016

I understand and have adhered to all the tenets of the Duke Community Standard in completing every part of this assignment. I understand that a violation of any part of the Standard on any part of this assignment can result in failure of this assignment, failure of this course, and/or suspension from Duke University.

## Contents



## List of Figures

# 1 Palm Figure 6.1-2

I gave x equally spaced values between 0 and 4 by using linspace code. Then I graphed [2, p. 265];

$$y = x^m \tag{1}$$

for different values of m(-0.5,0,0.5,1,2). I labeled each curve for different values of m with the m value, and gave title to the plot.

# 2 Chapra Problem 3.5

For the Chapra Problem 3.5, my program worked with different values of angle. With increasing number of N(number of steps used in Taylor Series), the accuracy for the given angle increased. On the other hand, with increasing angles, the accuracy of approximations decreased meaning the error increased. Even if the angles are actually are same, meaning two pi subtracted from the angle is same, bigger corresponding angle gave a higher error.

# 3 Chapra Problem 4.1

Table of Maclaurin series [1, p. 120] is;

| $a$ | $\epsilon_s$ | maxit | $\sqrt{a}$ | $\epsilon_a$ | iter |
|---|---|---|---|---|---|
| 16 | 1e-02 | 5 | 4.002257524798522e+00 | 3.36e+00 | 5 |
| 16 | 1e-02 | 12 | 4.000000000000051e+00 | 1.59e-05 | 7 |
| 16 | 1e-08 | 5 | 4.002257524798522e+00 | 3.36e+00 | 5 |
| 16 | 1e-08 | 12 | 4.000000000000000e+00 | 1.27e-12 | 8 |
| 160 | 1e-02 | 5 | 1.482664109800340e+01 | 5.22e+01 | 5 |
| 160 | 1e-02 | 12 | 1.264911068004731e+01 | 7.89e-03 | 8 |
| 160 | 1e-08 | 5 | 1.482664109800340e+01 | 5.22e+01 | 5 |
| 160 | 1e-08 | 12 | 1.264911064067352e+01 | 0.00e+00 | 10 |
| 1600 | 1e-02 | 5 | 1.052575377021292e+02 | 9.25e+01 | 5 |
| 1600 | 1e-02 | 12 | 4.000000000060651e+01 | 5.51e-04 | 10 |
| 1600 | 1e-08 | 5 | 1.052575377021292e+02 | 9.25e+01 | 5 |
| 1600 | 1e-08 | 12 | 4.000000000000000e+01 | 1.52e-09 | 11 |
| 16000 | 1e-02 | 5 | 1.005306930179416e+03 | 9.92e+01 | 5 |
| 16000 | 1e-02 | 12 | 1.264911064067374e+02 | 1.86e-05 | 12 |
| 16000 | 1e-08 | 5 | 1.005306930179416e+03 | 9.92e+01 | 5 |
| 16000 | 1e-08 | 12 | 1.264911064067374e+02 | 1.86e-05 | 12 |
| 160000 | 1e-02 | 5 | 1.000531194227066e+04 | 9.99e+01 | 5 |
| 160000 | 1e-02 | 12 | 4.000285706905372e+02 | 1.20e+00 | 12 |
| 160000 | 1e-08 | 5 | 1.000531194227066e+04 | 9.99e+01 | 5 |
| 160000 | 1e-08 | 12 | 4.000285706905372e+02 | 1.20e+00 | 12 |

The quality of approximations increases as the iteration number increase. For the given iteration number, smaller values of $a$ have higher quality of approximations. This can also be observed from increasing number of iterations as $a$ increases. While for small values of $a$ expected accuracy can be reached before iterations have reached to their maximum value, for higher values of $a$ iterations reaches the maxvalue before they can reach to expected accuracy.

# 4 3D Projections

The figure and codes are in the appropriate appendices.

# 5 Data Logger

The diary, data file, and code are in the appropriate appendices.

# A Codes

## A.1 ExponentialsGraph.m

```
%[ExponentialsGraph.m]
%[Cemal Yagcioglu]
%[October 2,2016]-
% I have adhered to all the tenets of the
% Duke Community Standard in creating this code.
% Signed: [cy111]
x = linspace(0,4);

for m=[-0.5,0,0.5,1,2]
    axis([0,4,0,4])
    hold on
    y=x.^m;
    xticks([0 0.5 1 1.5 2 2.5 3 3.5 4])
    plot(x,y,'k')


end
hold off
 text(2.75,0.4,'\it{m} = -0.5')
 text(3.38,1.2,'\it{m} = 0')
 text(3.3,2.1,'\it{m} = 0.5')
 text(2.75,3.2,'\it{m} = 1')
 text(1.5,3.25,'\it{m} = 2')
 xlabel('\it{x}')
 ylabel('\it{y}')
 title('The Power Function \it{y = x^m} (cy111)')

 print -depsc ThePowerFunctionPlot

```

## A.2 CosSeries.m

```
%[CosSeries.m]
%[Cemal Yagcioglu]
%[October 2,2016]
% I have adhered to all the tenets of the
% Duke Community Standard in creating this code.
% Signed: [cy111]

function [CosineApprox,Error]=CosSeries(x,N)
CosineApprox(1) = 1
for k=2:N %1 is defined, starting from 2
    CosineApprox(k)=CosineApprox(k-1)+(x.^(2.*(k-1)).*((-1).^(k+1))./factorial(2.*(k-1)))
end
CosineApprox(1)=1;
Error=(((cos(x)-CosineApprox)./cos(x)).*100) %calculates error percent
end

```

## A.3 DivAvg.m

```
function [fx, ea, iter] = DivAvg(a, es, maxit)
```

```
% DivAvg  Use Divide and Average to find square root
%   [fx, ea, iter] = DivAvg(a, es, maxit)
%     a: number of which to take the square root
%     es: stopping error
%     maxit: maximum number of iterations
%     fx: approximation of square root of a
%     ea: approximate relative error (%)
%     iter: number of iterations
% Based on IterMeth.m from Figure 4.2 on p. 94 of
% Applied Numerical Methods with MATLAB for
% Scientists and Engineers
% Steven C. Chapra, 3rd Edition

%[DivAvg.m]
%[Cemal Yagcioglu]
%[October 2,2016]
% I have adhered to all the tenets of the
% Duke Community Standard in creating this code.
% Signed: [cy111]

%% defaults:
if nargin<2||isempty(es),es=0.0001;end
if nargin<3||isempty(maxit),maxit=50;end
%% initialization
iter = 1; sol = 1; ea = 100;
%% iterative calculation
while (1)
solold = sol;
sol = (sol + a./(sol))/(2);
iter = iter + 1;
if sol~=0
ea=abs((sol - solold)/sol)*100; end
if ea<=es || iter>=maxit,break,end
end
fx = sol;









end
```

## A.4 ThreeDProjections.m

```
%[ThreeDProjections.m]
%[Cemal Yagcioglu]
%[October 2,2016]
% I have adhered to all the tenets of the
% Duke Community Standard in creating this code.
% Signed: [cy111]

t = linspace(-10,10,1000);
```

```
x = ((10.*exp(-0.5.*t)).*sin(3.*t+2));
y = (7.*exp(-0.4.*t)).*cos(5.*t-3);
%x=t.*cos(t)
%y=t.*sin(t)
z = ((8.*t./5)-8);


subplot(2,2,1)
plot(x,y)
axis equal;
axis([-10,10,-10,10])
grid on
xlabel('x')
ylabel('y')

subplot(2,2,2)
plot3(x,y,z)
axis equal;
axis([-10,10,-10,10,-10,10])
view(45, 35)
grid on
xlabel('x')
ylabel('y')
zlabel('z')
title('Parametric Plots (cy111)')
hold off

subplot(2,2,3)
plot(x,z)
axis equal;
axis([-10,10,-10,10])
grid on
ylabel('z')
xlabel('x')


subplot(2,2,4)
plot(y,z)
axis equal;
axis([-10,10,-10,10])
grid on
ylabel('y')
xlabel('z')

print -deps ParametricPlots
```

## A.5 DataLogger.m

```

  %delete('MyTemps.tex')
Temp = [];
i=1;
Max=0;
Min=999999;
AvTemp=0;
while i<99999999
```

```
    FID = fopen('MyTemps.txt', 'w');
    TempInput2 ='%4.2f';
    fprintf(FID,TempInput2,Temp);
     Temp(i) = input('Enter a Temperature: ');
     T=Temp(i);
    if T >= 0
    OldTemp = Temp;
    AvTemp = ((AvTemp*(i-1))+T)/(i);
    if T > Max
        Max = T;
    end
    if T < Min
        Min = T;
    end
TempInput ='Enter a Temperature:  %4.2f\n';
fprintf(TempInput,T);
fprintf('Readings Minimum Average Maximum\n');
formatSpec = '       %0.0f  %4.2f  %4.2f  %4.2f\n';
fprintf(formatSpec,i,Min,AvTemp,Max);
    else
%TempInput ='Enter a Temperature:  %4.2f\n';
%fprintf(TempInput,Temp);
        i=i+1000000000;
    end
    i=i+1;
    fprintf(FID, '\n');




end
save MyTemps.txt Temp -ascii
```

# B Diary and Data Sets

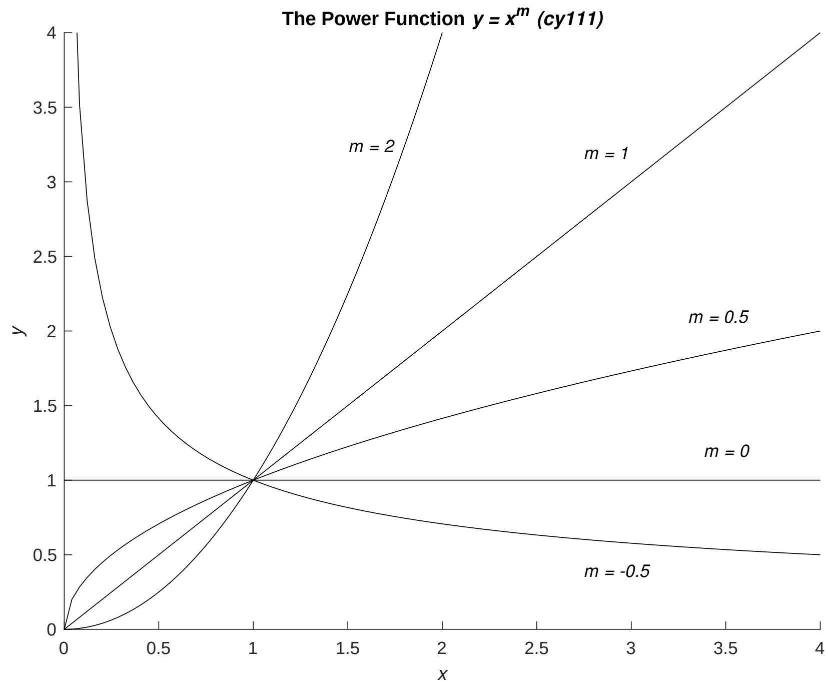
## B.1 MyTemps.txt

```
   3.0000000e+02   2.0000000e+02   2.5000000e+02  -1.0000000e+00
```

## B.2 TempDiary.txt

```
out =
  259.0713
Enter a Temperature:  259.07
Readings Minimum Average Maximum
       1  259.07  259.07  259.07
out =
  262.9039
Enter a Temperature:  262.90
Readings Minimum Average Maximum
       2  259.07  260.99  262.90
out =
  254.2658
Enter a Temperature:  254.27
Readings Minimum Average Maximum
       3  254.27  258.75  262.90
out =
  320.9484
Enter a Temperature:  320.95
Readings Minimum Average Maximum
       4  254.27  274.30  320.95
out =
  256.4189
Enter a Temperature:  256.42
Readings Minimum Average Maximum
       5  254.27  270.72  320.95
out =
  270.1800
Enter a Temperature:  270.18
Readings Minimum Average Maximum
       6  254.27  270.63  320.95
```

# C Figures



The Power Function $y = x^m$ (cy111)
m = 2
m = 1
m = 0.5
m = 0
m = -0.5
y
x
0
0.5
1
1.5
2
2.5
3
3.5
4

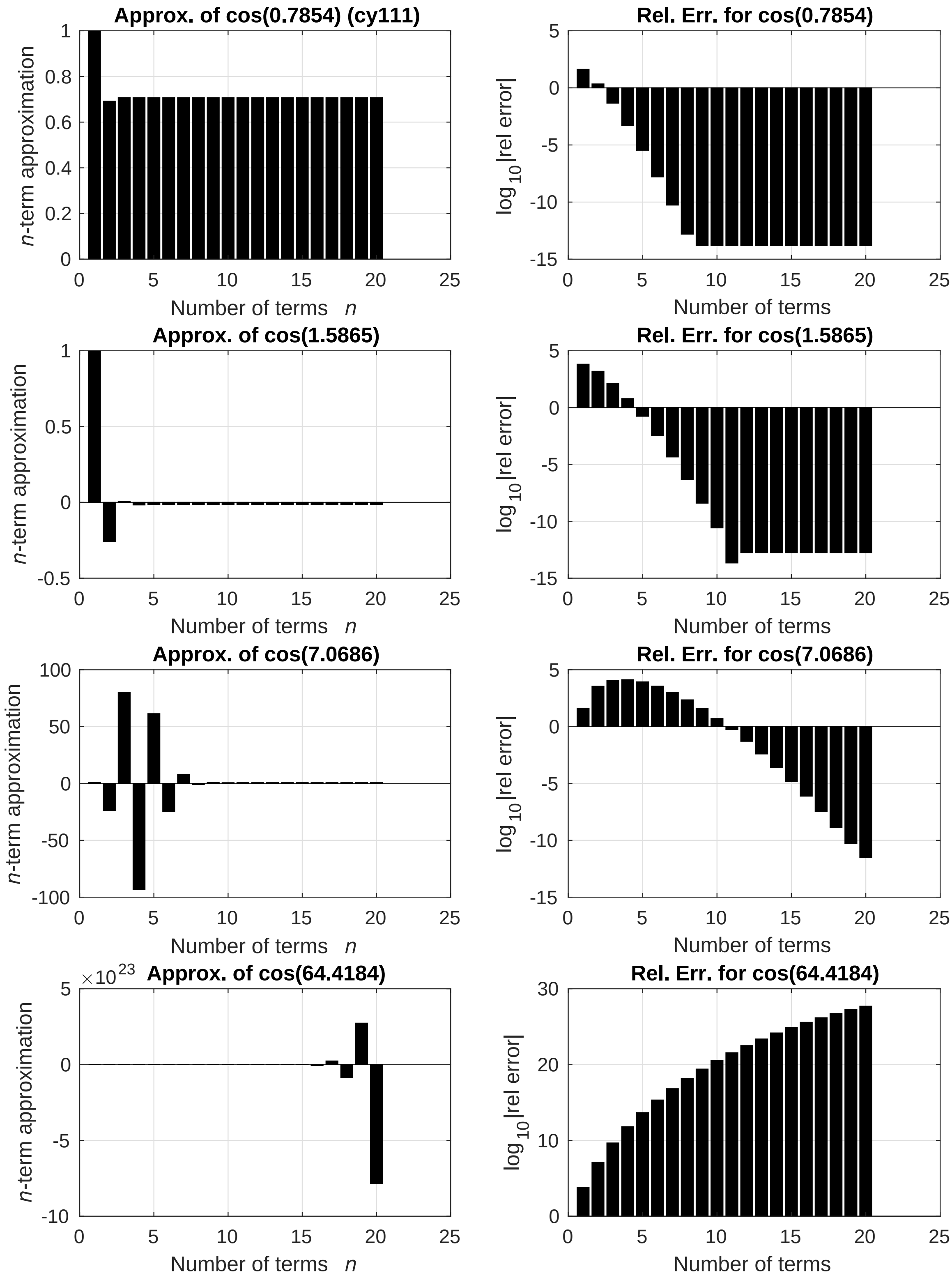



Figure 1: Output of `CosSeriesChecker.m` for Chapra 3.5

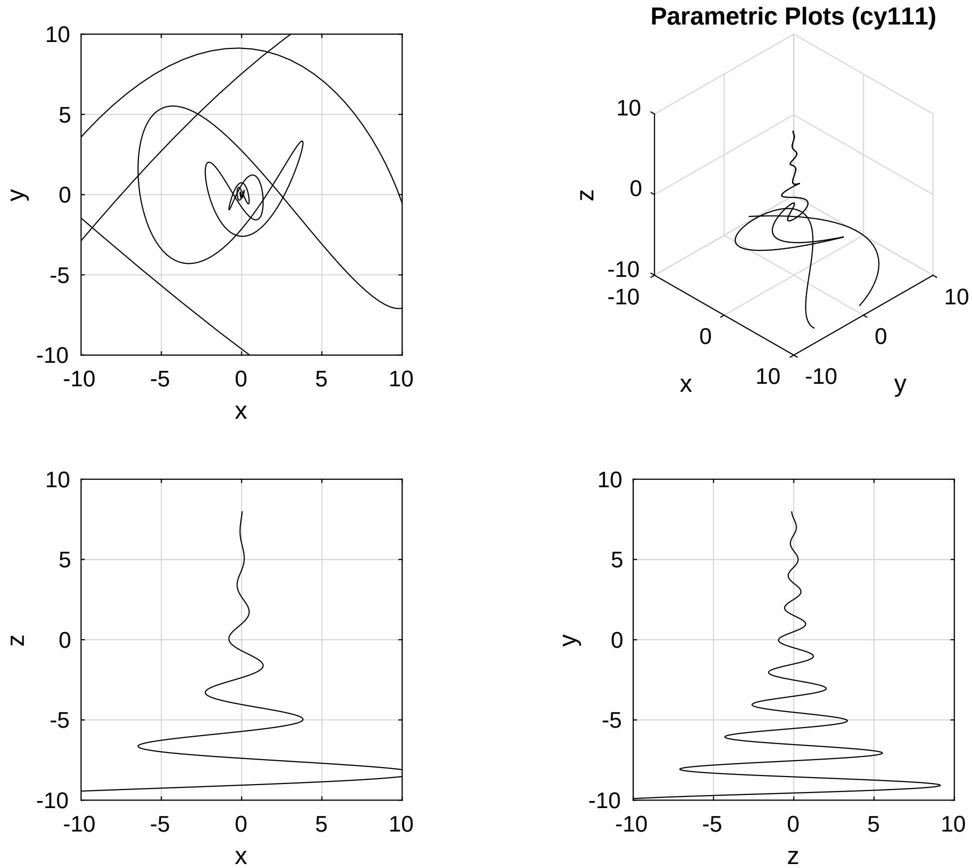



Figure 2: Output of `ParametricPlots.m` for Palm 5.15

# References


[1] Chapra, Steven C., *Applied Numerical Methods with MATLAB for Engineering and Scientists*. McGraw-Hill, New York, 3rd Edition, 2012.

[2] Palm, William J., *Introduction to MATLAB for Engineers*. McGraw-Hill, New York, 3rd Edition, 2011.