Lab 11

ENED 1090: MODELS I
Week 11 Laboratory

Submit Week 12 during Lab

姓名:	易弘睿
Xìngmíng (拼音):	Yi Hongrui
English Name:	Horace
CQU Student ID:	20186103

INSTRUCTIONS

Complete each question below by typing your answer or copying from the output in MATLAB or Excel.

This assignment is to be completed outside of class. You will submit a digital copy to your TA during the lab session next week.

!!! **To receive points for this assignment, add your name to the filename. For example, if my name is Lin Yali, I will change the filename to**

Wk11_ened1090_laboratory_LinYali.doc

OBJECTIVES

For this assignment, students will demonstrate

- Arrays

For this assignment,: you will apply the instructions to MATLAB (or Octave).
Each problem will refer the Week 10 – If you are working on these programs, use the Data files that are provided on Blackboard.

How to use the data files:

1. *put the *.mat file in your current director.*
2. *create a new script, and complete your program header*
3. *add the **clear; clc; close all** commands*
4. *use the **load** command to import the *.mat file*
   a. *i.e. typing **load RandomNumberDistribution_randiData** will load the data for the first problem*
5. *happy programing the graph ☺*

FOR ALL PROBLEMS:

To receive full credit on each plot. You will need to include

- labels on x, y and z axis
- legend describing each curve (2D plot)
- labels on the colorbar (3D plot)
- title

PROBLEM 1

Consider the Statistics Problem from Wk 10: **RandomNumberDistribution**

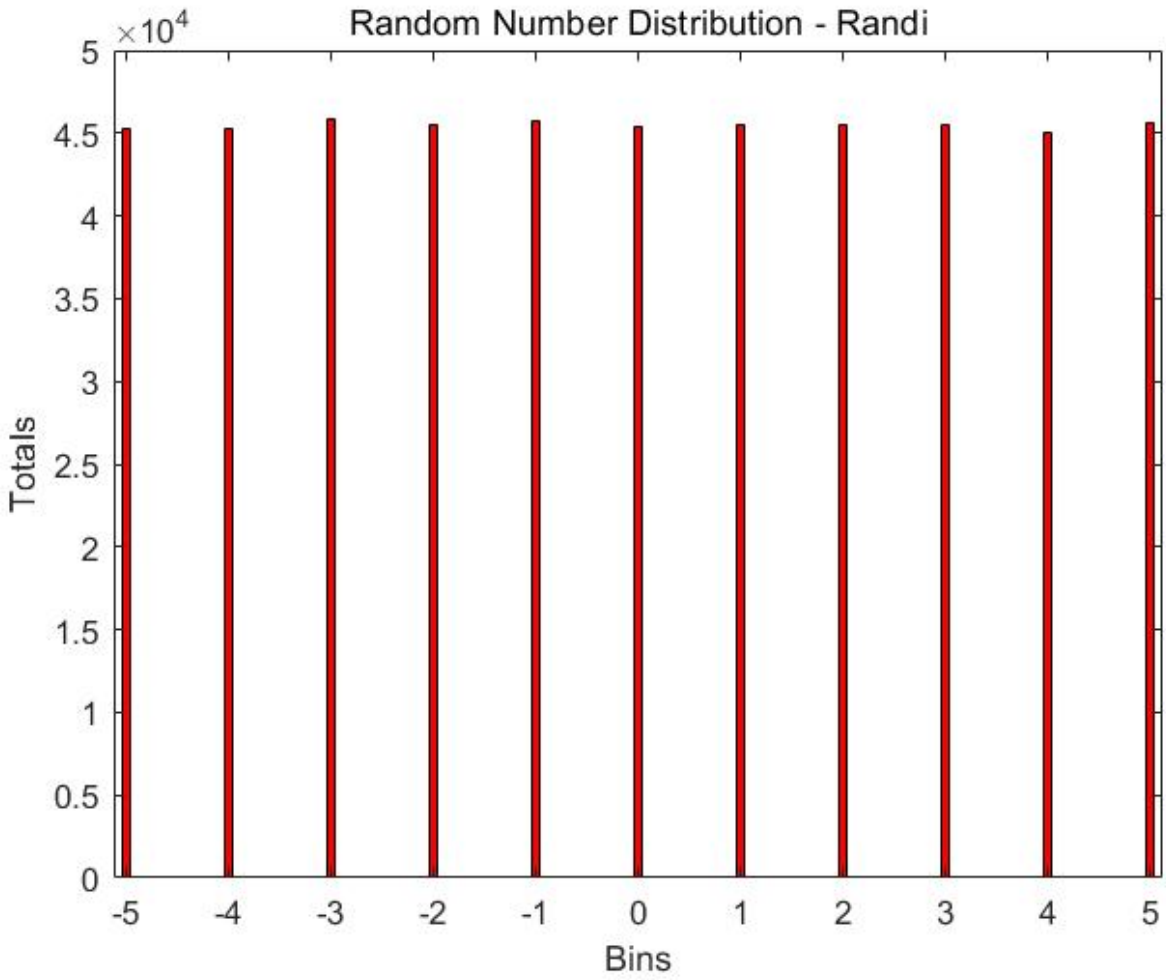
Using your program or the data files: **RandomNumberDistribution_randiData** and **RandomNumberDistribution_randnData**,

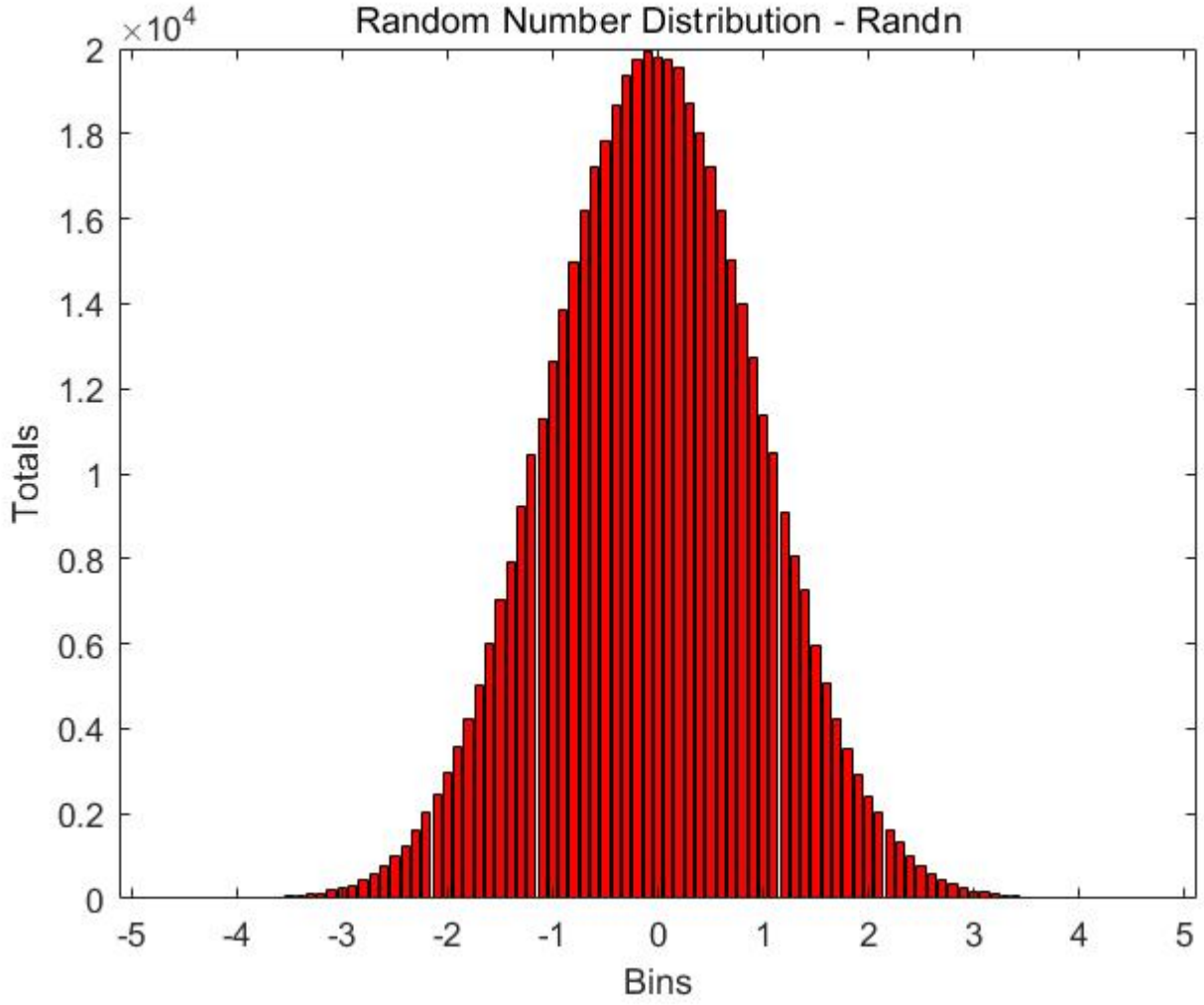
- Add a graph that gives a meaningful display to the problem.

Some suggestions:

- **Make sure you add close all; at the beginning of your code**

Copy an **image** of the graph for the two types of random numbers


Random Number Distribution - Randi
×10^4
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Totals
-5
-4
-3
-2
-1
0
1
2
3
4
5
Bins


Random Number Distribution - Randn
×10^4
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
Totals
-5
-4
-3
-2
-1
0
1
2
3
4
5
Bins

Copy the code *of just your plot block*.

```
%% RandomNumberDistribution
% Name : Horace
% Date : 17 Nov 2018
% Description : a demonstration on plotting methods

%% CODE
clear; clc; close all;

load RandomNumberDistribution_randiData.mat

figure(1)
bar(bins,[total N-sum(total)],'r')
xlabel('Bins')
ylabel('Totals')
title('Random Number Distribution - Randi')

clear;

load RandomNumberDistribution_randnData.mat

figure(2)
bar(bins,[total N-sum(total)],'r')
xlabel('Bins')
ylabel('Totals')
title('Random Number Distribution - Randn')
```

Bonus (+5 Points)

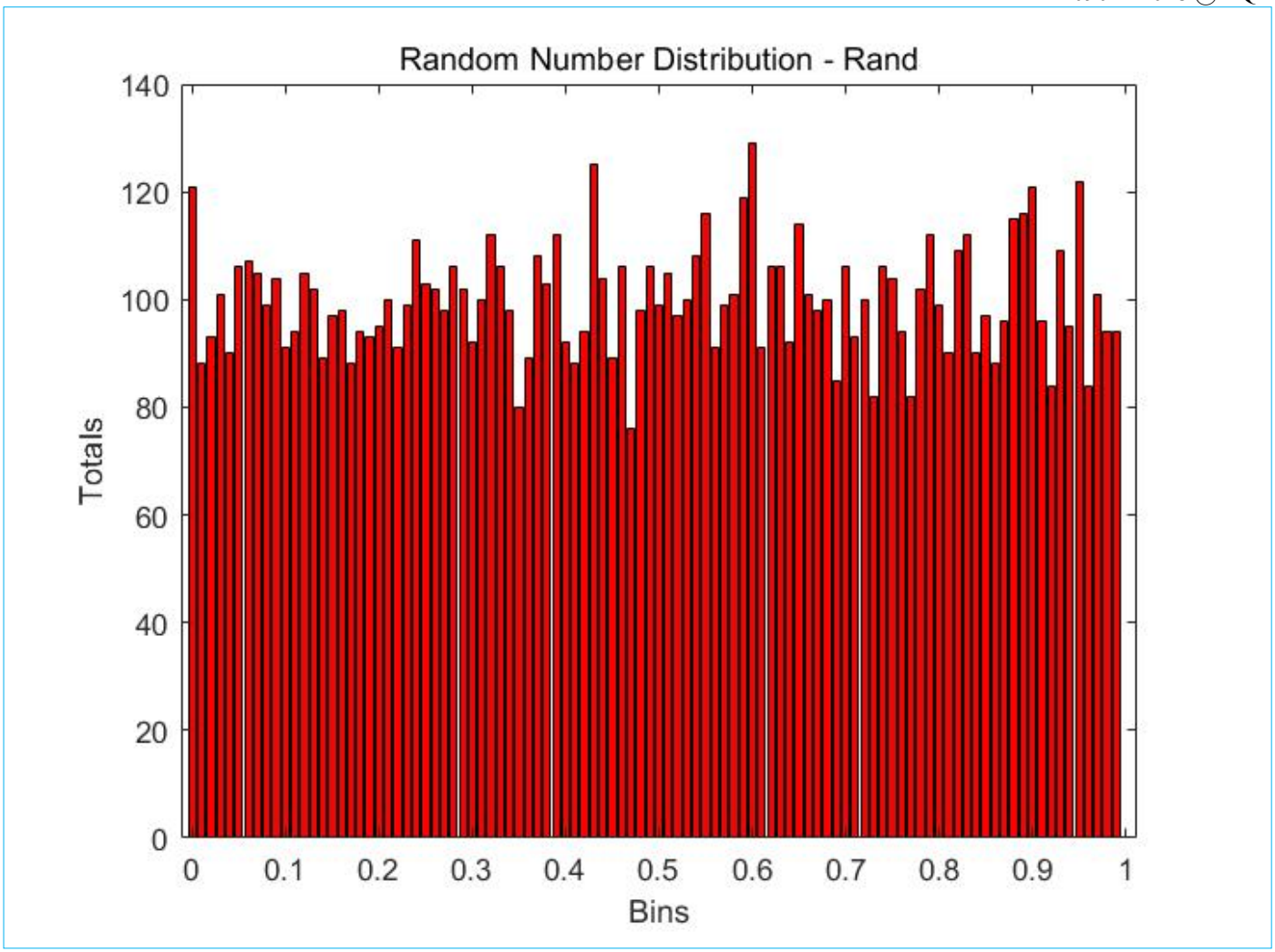
A final random number generator in MATLAB is the **rand** function.

- Modify your code to include a new option for **numbers = rand(1,N)**

Based on your graph for this command, what does **rand** do?

Get a random matrix of 1 rows and N columns less than 1.

Copy an **image** of the graph for this third case.



PROBLEM 2
Consider the Biology Problem from Wk 10: **PredatorPrey**

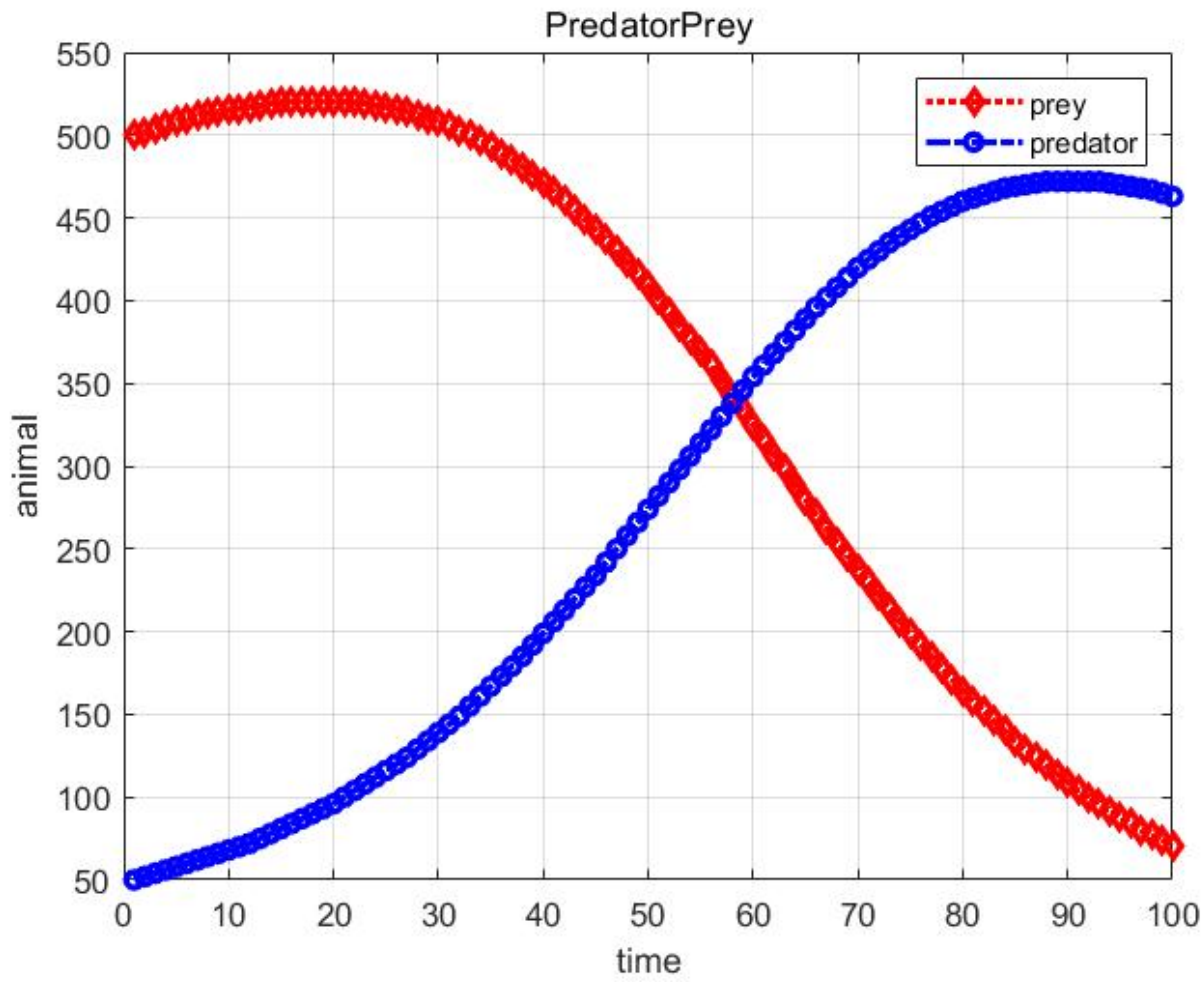
Using your program or the data files: **PredatorPrey_Data**,

- Add a graph that gives a meaningful display to the problem.

Some suggestions:

- **Make sure you add close all; at the beginning of your code**

Copy an **image** of the graph showing the population of the two species over time



Increase the number of years in the simulation and try different values of A, B, C, and D.

- What is the effect of A, B, C, and D, on the survival of the two animals?

As A increase and B decrease, the survival of the prey gets higher;
As C increase and D decrease, the survival of the predator gets lower.

Copy the code *of just your plot block*.

```
clear; clc; close all;
load PredatorPreyData.mat
time = 1:100;
figure(1)
plot(time,P(1,:),':rd','lineWidth',2,'markerSize',5)
hold on
plot(time,P(2,:),'-.bo','lineWidth',2,'markerSize',5)
xlabel('time')
ylabel('animal')
title('PredatorPrey')
legend('prey','predator')
grid on
```

PROBLEM 3

Consider the Engineering Problem from Wk 10: **TemperaturePlate**

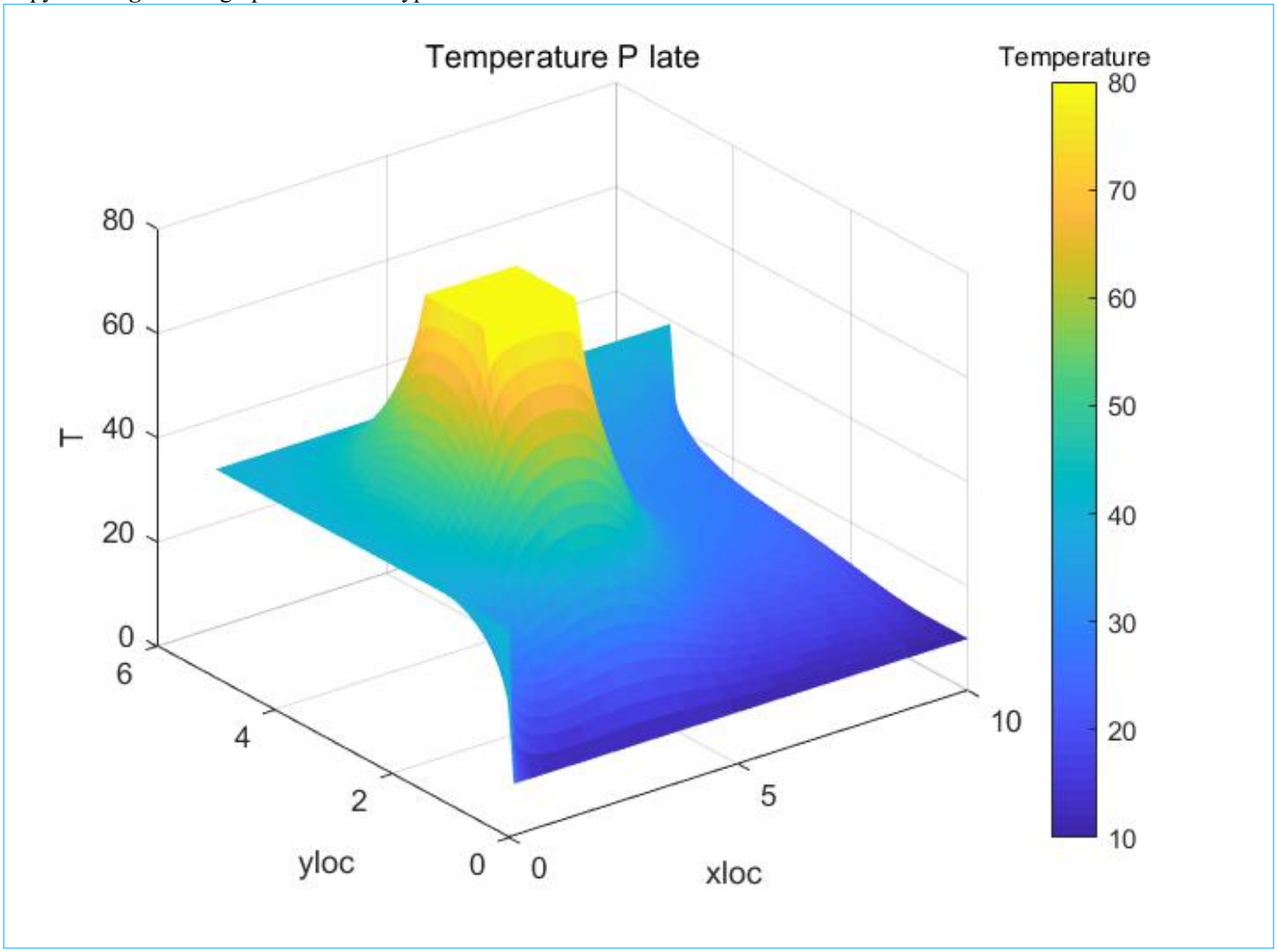
Using your program or the data files: **TemperaturePlate_Data**,

- Add a graph that gives a meaningful display to the problem.

Some suggestions:

- **Make sure you add close all; at the beginning of your code**

Copy an **image** of the graph for the two types of random numbers



Consider the two options for programing the y array

- y = YM:-L:0
- y = 0:L:YM

How does this statement affect your graph?

Using these two statements will get two graphs which are up-side-down to each other..

Copy the code *of just your plot block*.

```
clear; clc; close all;
load TemperaturePlateData.mat
figure(1)
surf(xloc,yloc,T,'edgeColor','none')
hold on
xlabel('xloc')
ylabel('yloc')
zlabel('T')
title('Temperature P late')
title(colorbar,'Temperature')
grid on
```

PROBLEM 4

Consider the Mathematics Problem from Wk 10: **TelescopePSeries**

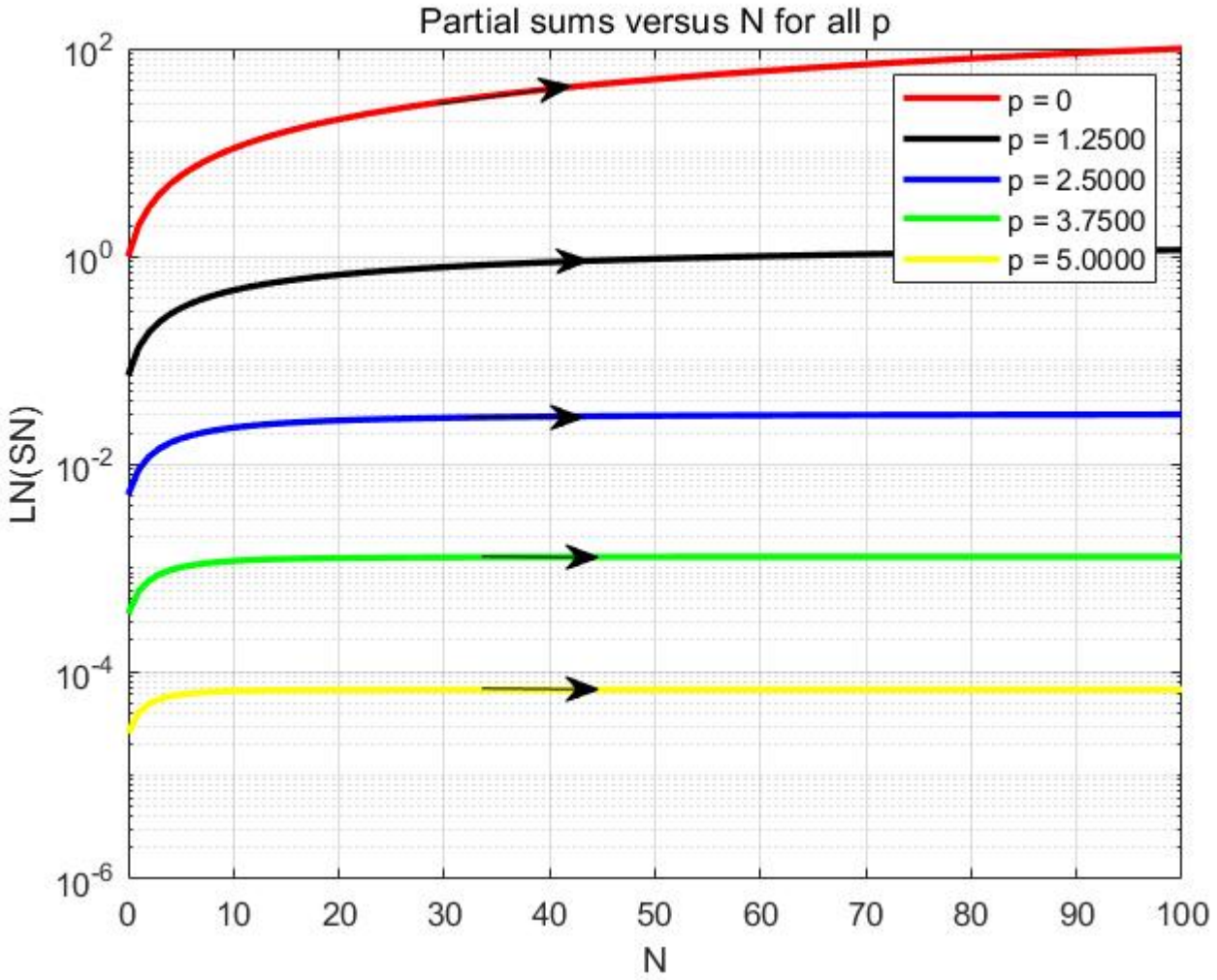
Using your program or the data files: **TelescopePSeries _Data**,

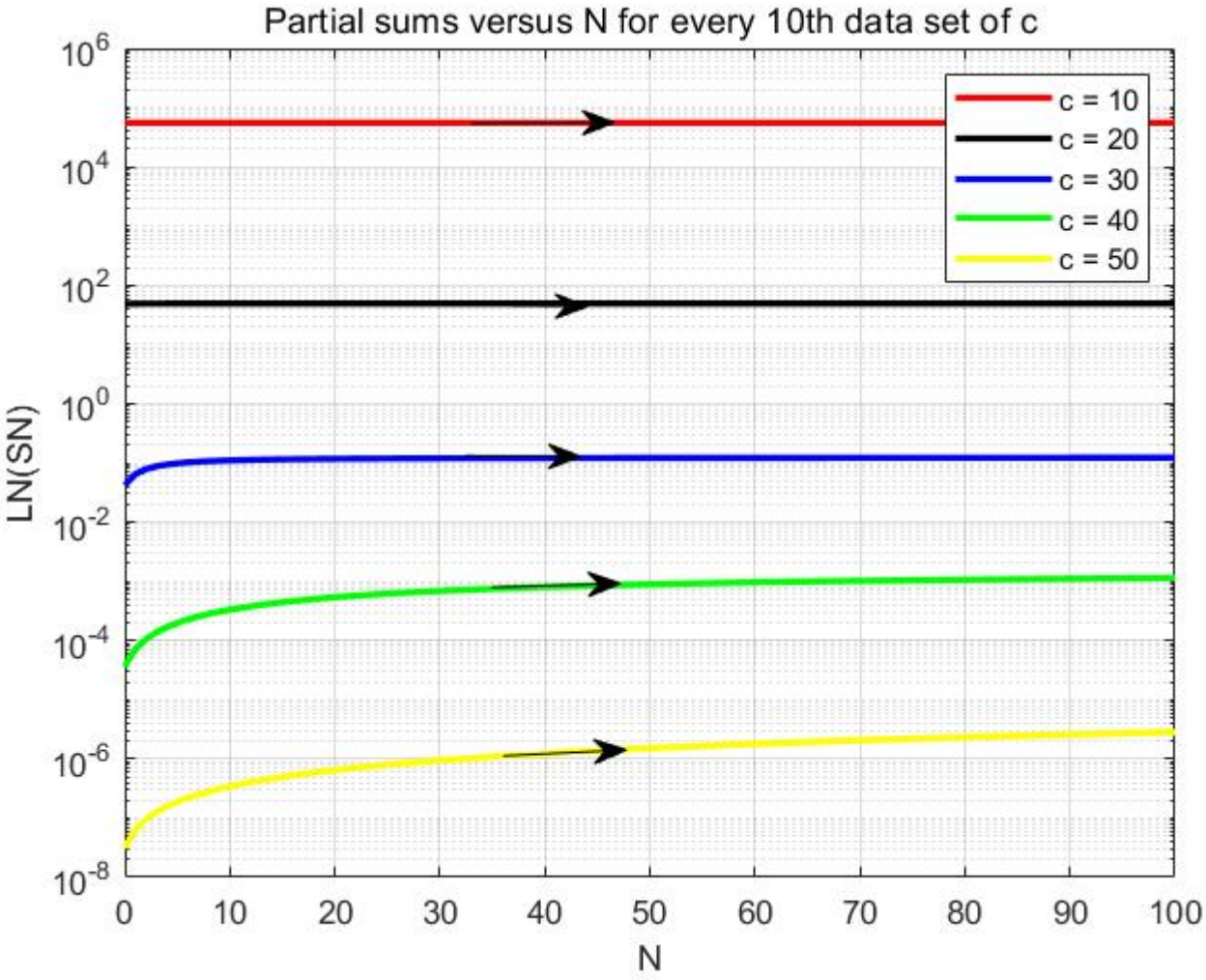
- Add graphs that gives a meaningful display to the problem
  - Using c(33), plot the partial sums versus N for all p
  - Using p(3), plot the partial sums versus N for every 10th data set of c
  - plot the partial sums versus c and p, for every 10th data set of N

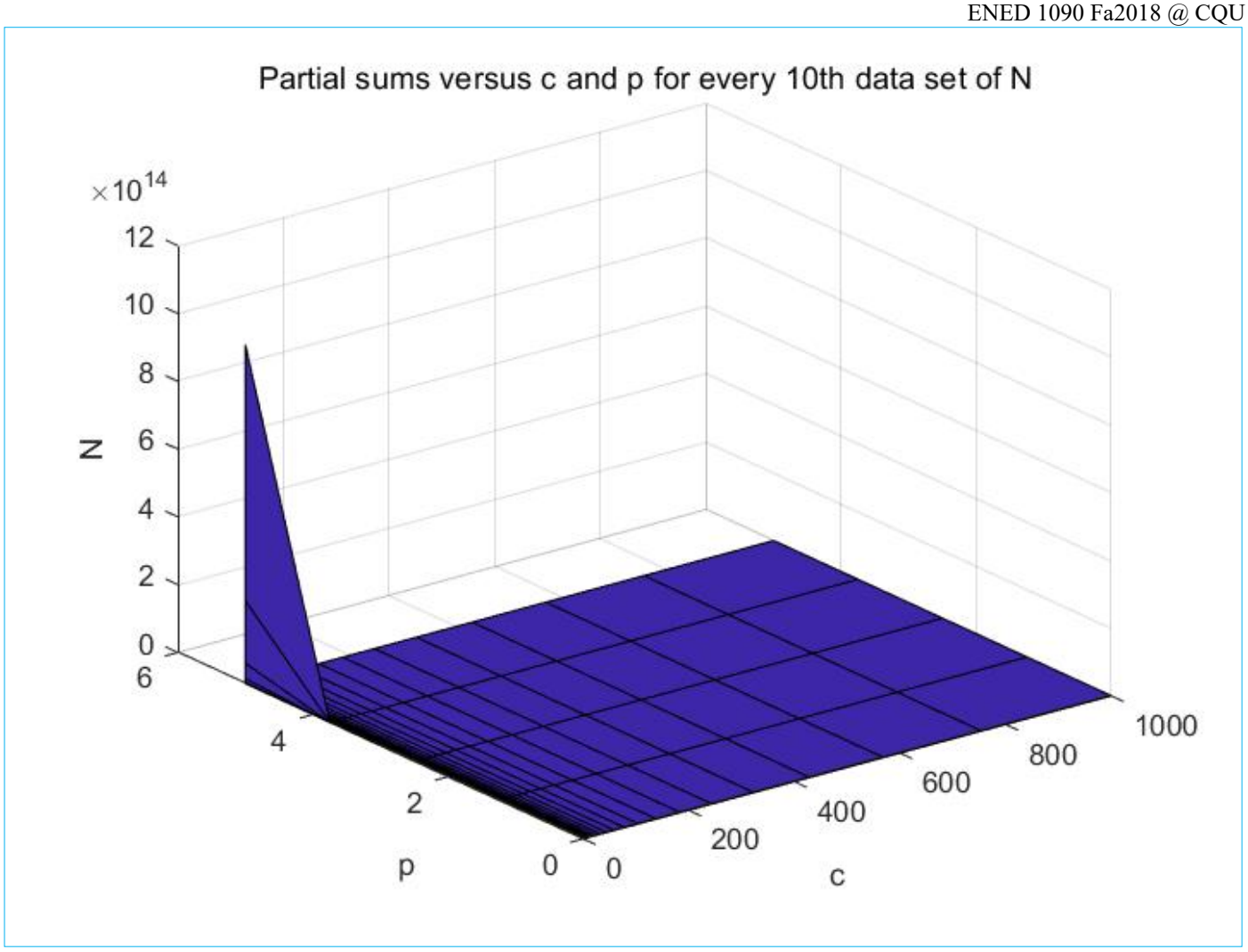
Some suggestions:

- **Make sure you add close all; at the beginning of your code**
- Consider using a semilog or a loglog plot
- Add an arrow to show the direction of increase c or p (for the first two plots)
- Depending on how you set up the array, you will need to use a **squeeze** command to create the plots – squeeze reduces the unit dimensions of a multi-dimensional array

Copy an **image** of each graph




Partial sums versus N for every 10th data set of c
LN(SN)
N
c = 10
c = 20
c = 30
c = 40
c = 50
10^6
10^4
10^2
10^0
10^-2
10^-4
10^-6
10^-8
0
10
20
30
40
50
60
70
80
90
100



Copy the code *of just your plot block.*

```
%% Code
clear;clc;close all;

load TelescopePSeriesData.mat

%Figure(1)
figure(1)
A = squeeze(SN(:,33,:));
semilogy(N,A(1,:),'-r','lineWidth',2)
hold on
semilogy(N,A(2,:),'-k','lineWidth',2)
semilogy(N,A(3,:),'-b','lineWidth',2)
semilogy(N,A(4,:),'-g','lineWidth',2)
semilogy(N,A(5,:),'-y','lineWidth',2)
xlabel('N')
ylabel('LN(SN)')
title('Partial sums versus N for all p')
legend('p = 0','p = 1.2500','p = 2.5000','p = 3.7500 ','p =
5.0000' )
grid on

%Figure(2)
```

```
figure(2)
B = squeeze(SN(3,:,:));
semilogy(N,B(10,:),'-r','lineWidth',2)
hold on
semilogy(N,B(20,:),'-k','lineWidth',2)
semilogy(N,B(30,:),'-b','lineWidth',2)
semilogy(N,B(40,:),'-g','lineWidth',2)
semilogy(N,B(50,:),'-y','lineWidth',2)
xlabel('N')
ylabel('LN(SN)')
title('Partial sums versus N for every 10th data set of c')
legend('c = 10','c = 20','c = 30','c = 40 ','c = 50' )
grid on

%Figure(3)
figure(3)
C1 = squeeze(SN(:,:,1));
C2 = squeeze(SN(:,:,11));
C3 = squeeze(SN(:,:,21));
C4 = squeeze(SN(:,:,31));
C5 = squeeze(SN(:,:,41));
C6 = squeeze(SN(:,:,51));
C7 = squeeze(SN(:,:,61));
C8 = squeeze(SN(:,:,71));
C9 = squeeze(SN(:,:,81));
C10 = squeeze(SN(:,:,91));
C11 = squeeze(SN(:,:,101));
surf(c,p,C1)
hold on
surf(c,p,C2,'edgeColor','none')
surf(c,p,C3,'edgeColor','none')
surf(c,p,C4,'edgeColor','none')
surf(c,p,C5,'edgeColor','none')
surf(c,p,C6,'edgeColor','none')
surf(c,p,C7,'edgeColor','none')
title('Partial sums versus c and p, for every 10th data set of N')
```