STOCHASTIC PROCESS
LAB FILE

Submitted to:

Dr.Vivek Kumar Agarwal

Submitted by:

Apurv Agarwal

2K16/MC/019

INDEX

SNO.	EXPERIMENT	DATE	SIGNATURE
1.	Simulation of discrete parameter stochastic processes		
2.	Simulation of continuous parameter stochastic processes		
3.	Homogeneous and Non Homogenous Bernoulli Process		
4.	Homogenous, Non Homogenous Poisson Process and Renewal Process with Uniform Distribution		
5.	Simple Random Walk (Unrestricted)		
6.	Random walk with absorbing barriers		
7.	Random walk using Markov chains (absorbing and reflecting barriers)		
8.	Ergodic chain Steady state probabilities		
9.	M/M/1 Queuing Model		

EXPERIMENT # 1

QUESTION: Simulate the following discrete parameter stochastic processes

1) Discrete State Space: No. of cars washed on n^{th} day in a car wash given minimum 20 cars are washed and maximum 30 cars are washed.
2) Continuous State Space: Average time taken for a car to be washed on n^{th} day of month given time required is 2 minutes and maximum time taken is 4minutes.

CODE:

1) Discrete Parameter Discrete State Space –

```
x = [1:1:30];
y = 20 + randi([0 10],30,1);
p = scatter(x,y);
xlabel('Days of Month');
ylabel('No. of Car Washed on Nth Day');
title('Discrete Parameter - Discrete State Space');
```

2) Discrete Parameter Continuous State Space -

```
x = [1:1:30];
y = 120 + 120.*rand(30,1);
p = scatter(x,y);
xlabel('Days of Month');
ylabel('Avg Time Taken for Car Wash');
title('Discrete Parameter - Continuous State Space');
```

OUTPUT:

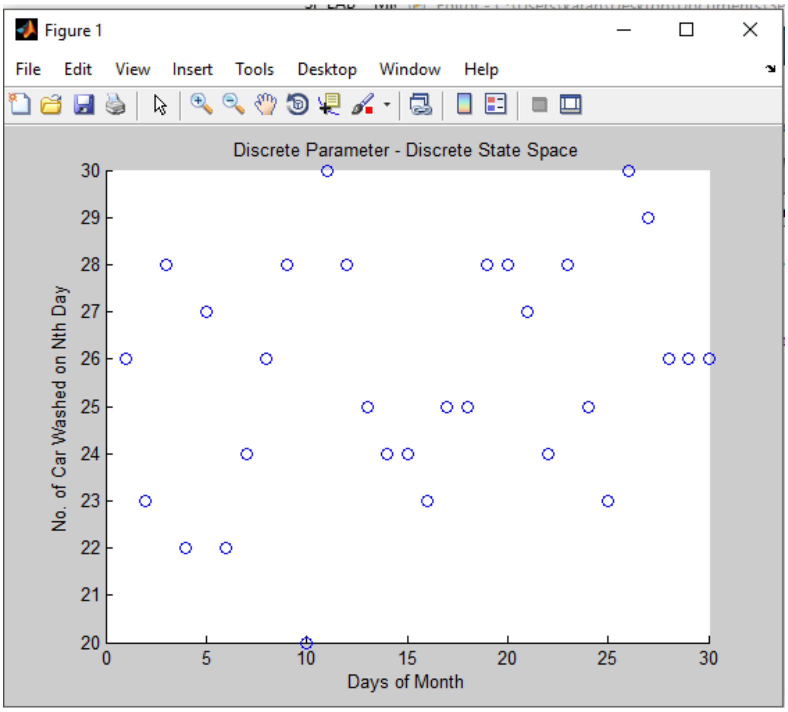
Figure 1
File Edit View Insert Tools Desktop Window Help
Discrete Parameter - Discrete State Space
No. of Car Washed on Nth Day
Days of Month
30
29
28
27
26
25
24
23
22
21
20
0
5
10
15
20
25
30

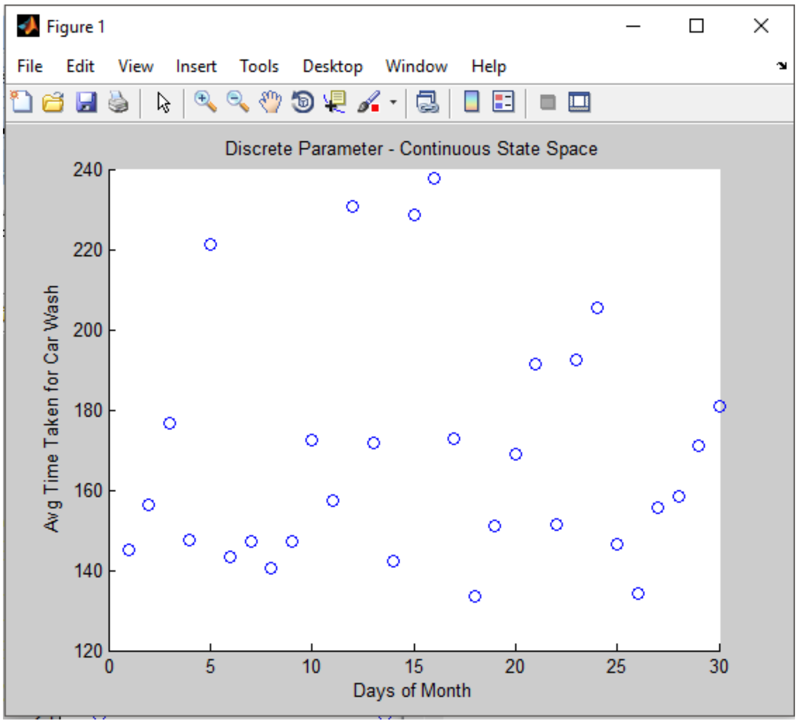
Figure 1
File Edit View Insert Tools Desktop Window Help
Discrete Parameter - Continuous State Space
Avg Time Taken for Car Wash
Days of Month
240
220
200
180
160
140
120
0
5
10
15
20
25
30

EXPERIMENT # 2

QUESTION: Simulate the following continuous parameter stochastic processes –

1) Continuous State Space: Variation in temperature in a time period of 5 minutes given that temperature can vary between 22 and 26 C.
2) Discrete State Space: No. of times temperature changed within 5 minutes and when it changed given temperature can change a maximum of 10 times.

CODE:

1) Continuous State Space:

```
x = [0.01:0.05:5];
y = 22 + 4.*rand(100,1);
p = plot(x,y);
xlabel('Time in minutes');
ylabel('Temp Change');
title('Continuous Parameter - Continuous State Space');
```

2) Discrete State Space:

```
x = [0.01:0.1:5];
y = randi([0 10],50,1);
p = stairs(x,y);
xlabel('Time in minutes');
ylabel('No. of times Temp Changed');
title('Continuous Parameter - Discrete State Space');
```

OUTPUT:

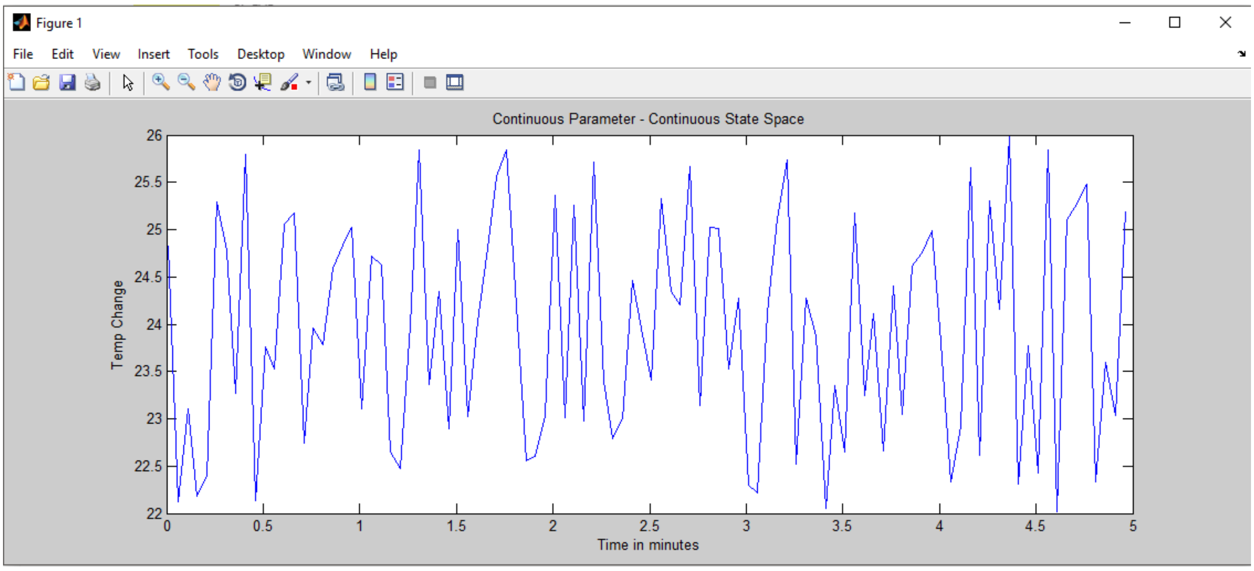

Figure 1
File Edit View Insert Tools Desktop Window Help
Continuous Parameter - Continuous State Space
Temp Change
Time in minutes
26
25.5
25
24.5
24
23.5
23
22.5
22
0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5

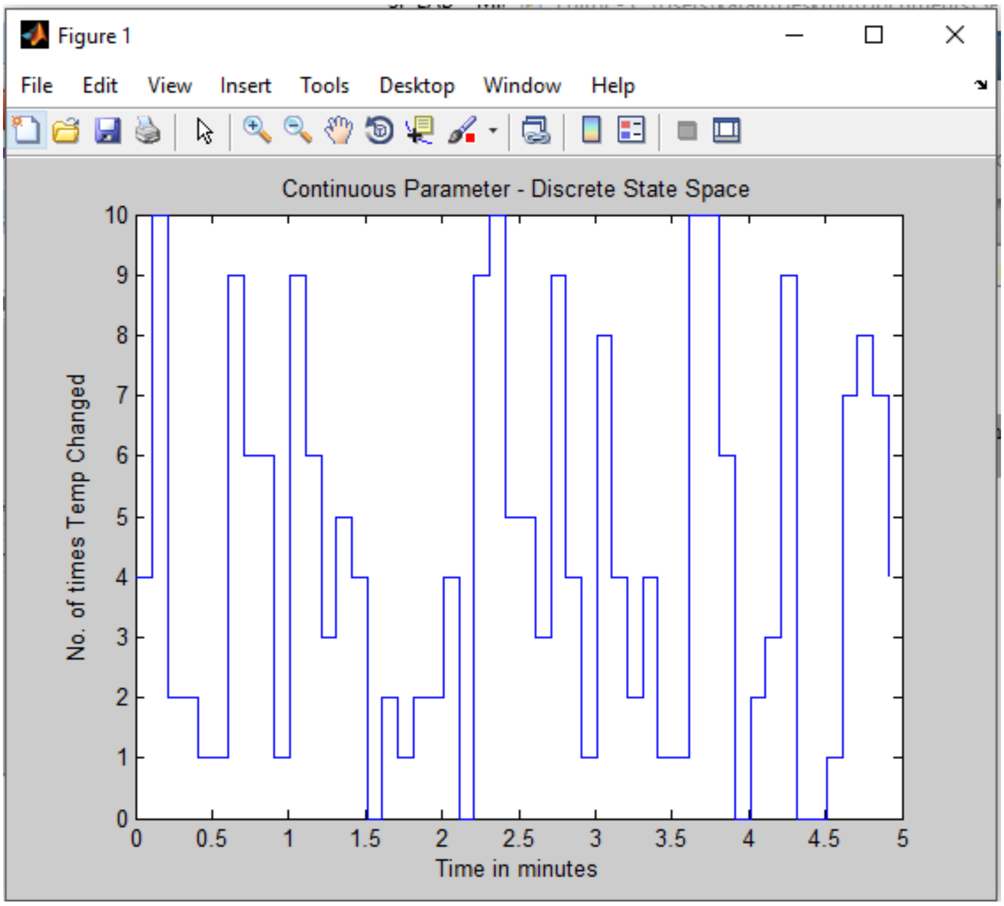

Figure 1
File Edit View Insert Tools Desktop Window Help
Continuous Parameter - Discrete State Space
No. of times Temp Changed
Time in minutes
10
9
8
7
6
5
4
3
2
1
0
0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5

EXPERIMENT # 3

QUESTION: It has been observed that a fuse designed by a company follows bernoulli distribution when being manufactured. If the company sells fuses in a box of 20 and gives guarantee that box will be replaced if no. of defective fuses is greater than 2. Find expected no. of replacements in a lot of 1000 boxes. Given -

1) Pr[fuse is defective] = 0.01 = p
2) Pr[n^{th} fuse is defective] = 0.01n = p_n

CODE:

Function file :

1) Homogenous Bernoulli

```
function [prob] = bernoulli(x,n)
p = 0.01;
q = 1-p;
prob = 0;
    for i=x+1:n
        prob = prob + (factorial(n)/(factorial(n-
i)*factorial(i)))*(p^i)*(q^(n-i));
    end
end
```

2) Non Homogenous Bernoulli

```
function [prob] = nbernoulli(x,n)
prob = 0;
p=1;
q=1;
    for i=x+1:n
        for j=1:i
            p = p*(0.01*j);
        end
        for j=i+1:n
            q = q*(1-(0.01*j));
        end
prob = prob + (factorial(n)/(factorial(n-i)*factorial(i)))*p*q;
        p=1;
        q=1;
    end
end
```

OUTPUT:

Command Window:

```
>> ans = 1000*bernoulli(2,20)

ans =

    1.0036

>> ans = 1000*nbernoulli(2,20)

ans =

    0.9193
```