Compute the bifurcation diagram for the logistic map. The logistic map is given by

$$x_{n+1} = \mu x_n (1 - x_n),$$

and the bifurcation diagram illustrates the behavior of the iterates of the map as a function of the parameter Here is a resonable outline:

Loop 1 Start at $\mu = 2.4$ and finish at $\mu = 4$.

Set $x = x_0$

Loop 2 Iterate logistic map a fixed number of times (transient)

Compute x

Loop 2 (end)

Loop 3 Iterate logistic map a fixed number of times (data)

Compute and save x

Loop 3 (end)

Loop 1 (end)

Warning #1: The array x_data will be graded. To pass the assessment, the indices and values in your x_dameans you are required to start your map iteration at x=x_0 and then iterate the map exactly n_trans times. another n_data times. The values of x you obtain from these second n_data iterations are to be stored in x match mine, the grader will not pass you.

Warning #2: You will need to write your iteration in the form $x=mu^*x^*(1-x)$. Roundoff errors and the butterfly systems will prevent you from passing the assessment if you write the equation in a different form, such as x = x + x + y + y = x + y + y = x + y + y = x + y =

Script @





MATLAB Documentation (https://www.mathworks.com/help/)

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14
```

```
mu_min=2.4; mu_max=4; %range of mu values
15
   n_mu=500; %number of mu pixels
16
   n_x=400; %number of x pixels
17
   mu_edges=linspace(mu_min,mu_max,n_mu+1); %edges of mu pixels
18
   mu=(mu_edges(1:n_mu)+mu_edges(2:n_mu+1))/2; %values of mu on which to perform compu
19
   x_edges=linspace(0,1,n_x+1); %edges of x pixels
20
21
   n_trans=20000; %transient iterations
22
   n_data=10000; %number of x values per mu value
23
24
   x_data=zeros(n_data,n_mu); %x-data used to construct figure
25
26
   x_0=0.5; %initial condition
27
28
   % WRITE THE COMPUTATIONAL ENGINE OF THE CODE.
29
   % USE THE ALREADY DEFINED PARAMETERS AND VARIABLES: n_mu, mu, x_0, n_trans, n_data.
30
   % YOUR FINAL RESULT WILL BE THE VARIABLE x_data, and this variable will be assessed
31
   for i = 1:size(mu,2)
32
       x = x_0;
33
       for j = 1:n_trans
34
           x = mu(i)*x*(1-x);
35
       end
36
       for j = 1:n_data
37
           x = mu(i)*x*(1-x);
38
           x_{data(j,i)} = x;
39
       end
40
   end
41
42
43
44
45
46
47
48
49
50
51
```

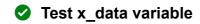
► Run Script

Assessment: All Tests Passed

Submit



0



Output



© 2022 The MathWorks, Inc.