Matlab script(hw13.m)

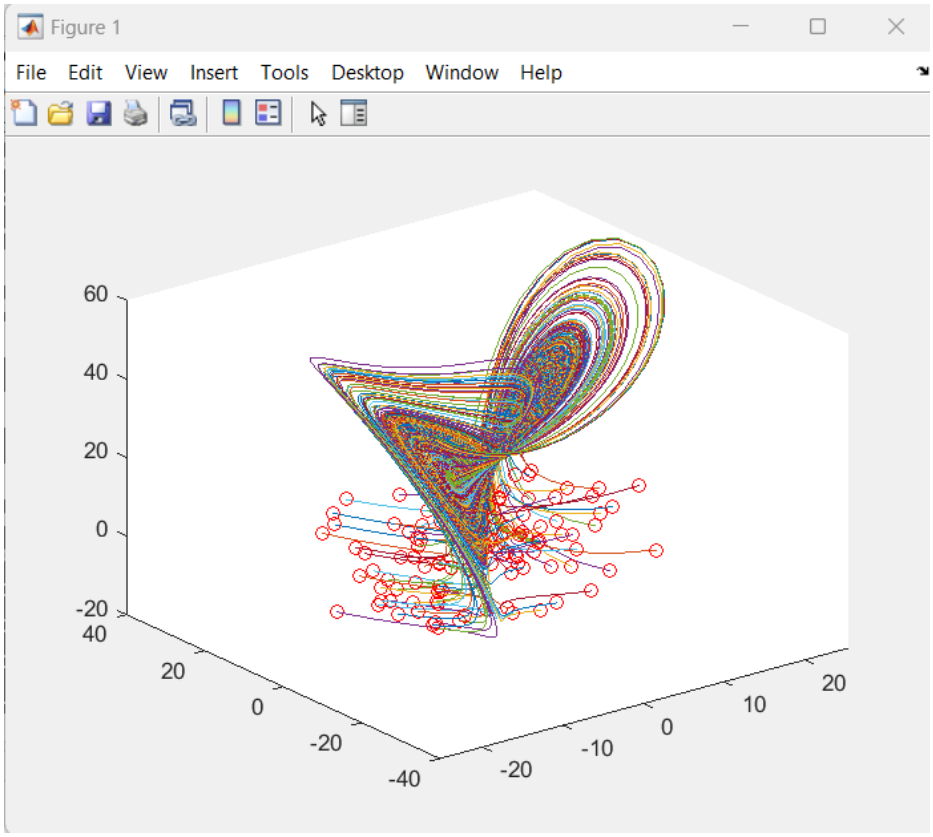
First steps:

I found a Loren-Oscillator script in the textbook(page 216 on my copy of it):

Code 6.7 Create training data of Lorenz trajectories.

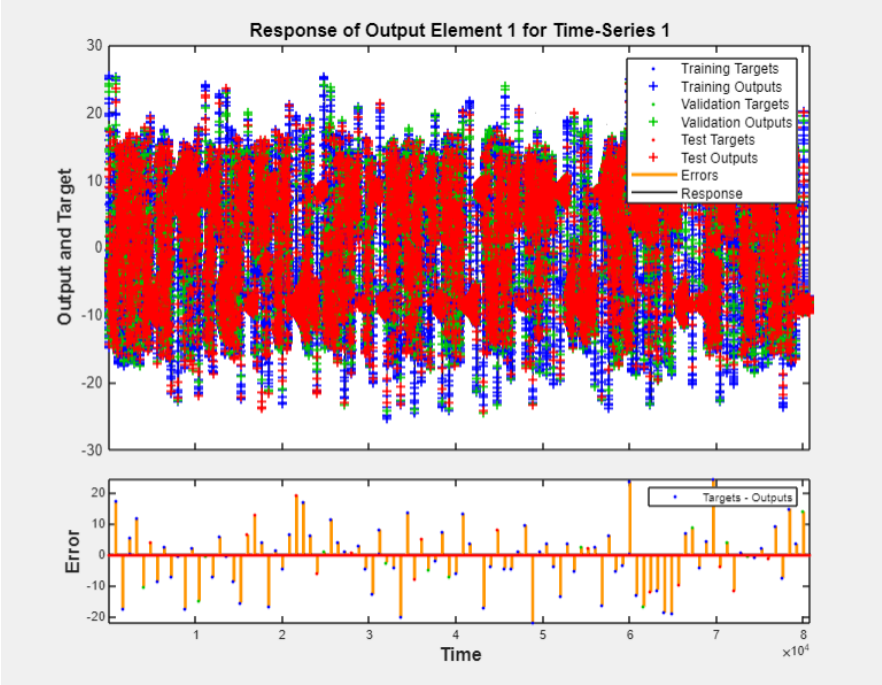
```
% Simulate Lorenz system
dt=0.01; T=8; t=0:dt:T;
b=8/3; sig=10; r=28;

Lorenz = @(t,x)([ sig * (x(2) - x(1))        ; ...
                  r * x(1)-x(1) * x(3) - x(2) ; ...
                  x(1) * x(2) - b*x(3)         ]);
ode_options = odeset('RelTol',1e-10, 'AbsTol',1e-11);

input=[]; output=[];
for j=1:100  % training trajectories
    x0=30*(rand(3,1)-0.5);
    [t,y] = ode45(Lorenz,t,x0);
    input=[input; y(1:end-1,:)];
    output=[output; y(2:end,:)];
    plot3(y(:,1),y(:,2),y(:,3)), hold on
    plot3(x0(1),x0(2),x0(3),'ro')
end
```

Which resulted this diagram:



Setting up NAR:

I basically just used the input data to train the NAR,



I then exported the code and modified so it plotted the prediction data:

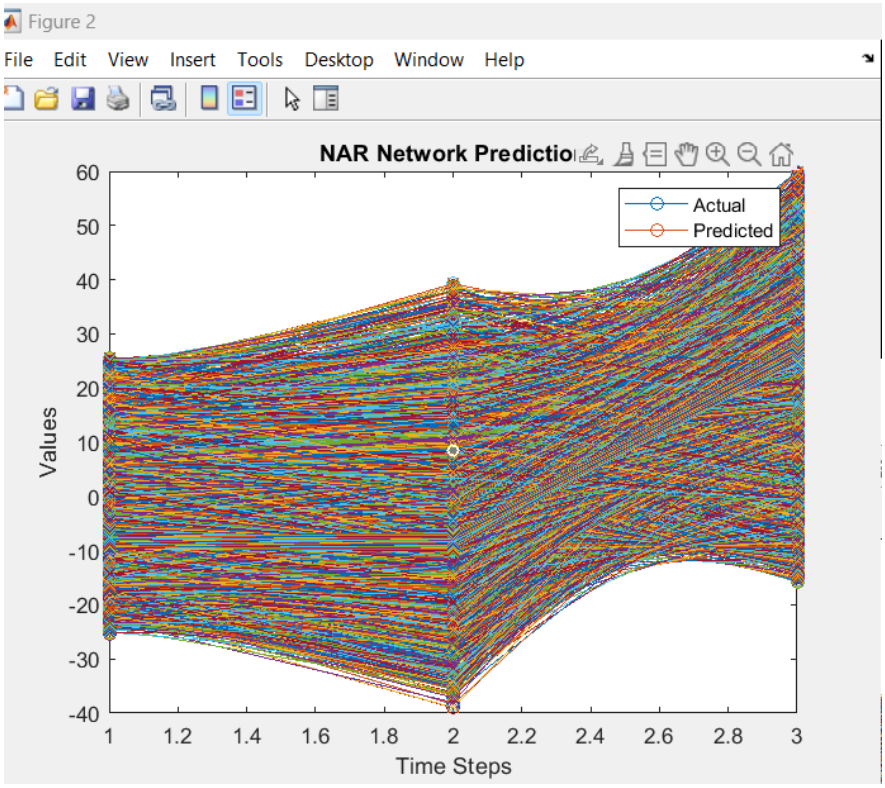


Figure 2
File Edit View Insert Tools Desktop Window Help
NAR Network Predictio
Actual
Predicted
Values
Time Steps
60
50
40
30
20
10
0
-10
-20
-30
-40
1
1.2
1.4
1.6
1.8
2
2.2
2.4
2.6
2.8
3