Modelling With Matlab – Assignment 1

1. Please see “holtderivs.m” for code.
2. Please see “holtsolve.m” for code. The strictly positive fixed point is:

1.0e+5\* [0.0010 , 0.0000 , 0.0023 , 0.0002 , 0.0003 , 0.0001 , 3.1561]

1. First it was necessary to make a time-dependent version of “holtderivs.m” which I called “tdholtdervis.m”, which is identical bar the fact it takes in (t,x). I then integrated with respect to time this function, ensuring I had enough iterations (2000), using a vector of random numbers as in parts 1 and 2. I then took the last iteration, and compared it with my fixed point in part 2.

[t,y] = ode45(@tdholtderivs, 0:2000, 100\*rand(1,7));

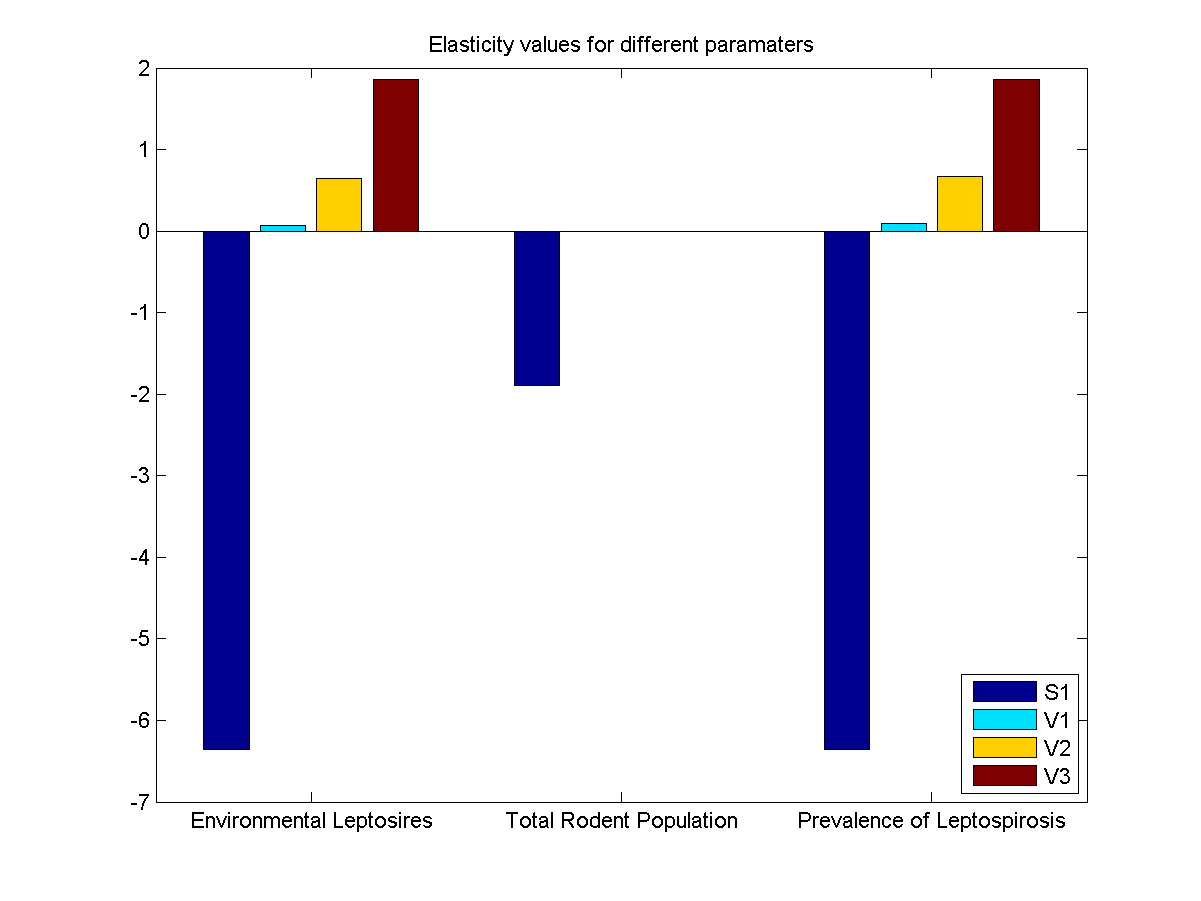
y(end, 1:7)

1.0e+05 \* [0.0010 , 0.0000 , 0.0023 , 0.0002 , 0.0003 , 0.0001 , 3.1583].

This is clearly converging to the fixed point in part 2, however more iterations may fix the last value to a greater precision.

1. To calculate the values for elasticity, it was easiest to create a near identical function to tdholterivs.m, but instead replace s1, v1, v2 and v3 with variables, and make the function take in an 11-vector instead of a 7-vector. This is called “elasticity.m”

On the graph can see the elasticity data for all 4 parameters, for the environmental leptospires, the total rodent population and the prevalence of leptospirosis. We can clearly see the magnitudes of the bars for s1 and v3 are much larger than those for v1 and v2. This shows s1 and v3 are much more important factors to the stability of the system. “elasticityall.m” computes each of the 12 elasticities and presents them in a bar chart.

1. I first wrote a function called “jack.m” in which I wrote the jacobian matrix for our system. It took the four variables s1,v1,v2,v3 computes the jacobian matrix with these parameters, evaluated at the fix point. “q5.m” is where I take those jacobian matrices, take the real part of the eigenvalue closest to zero, and compute the elasticity. The graph below show the values: [-15.5646 0.1329, 3.0827, 5.1540 ]. This

