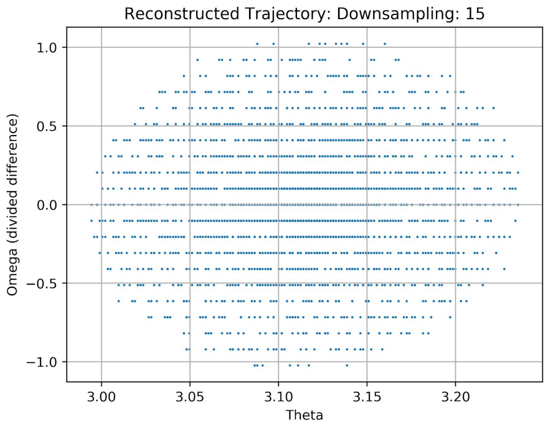
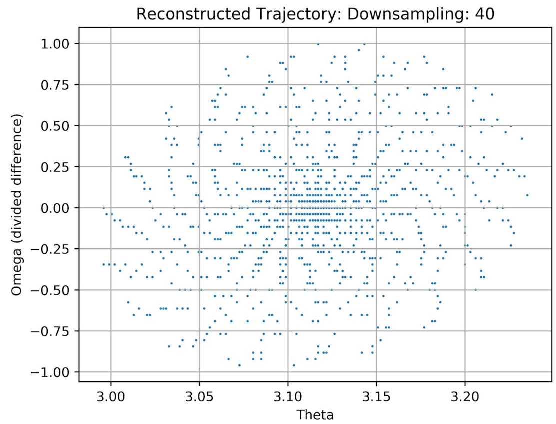
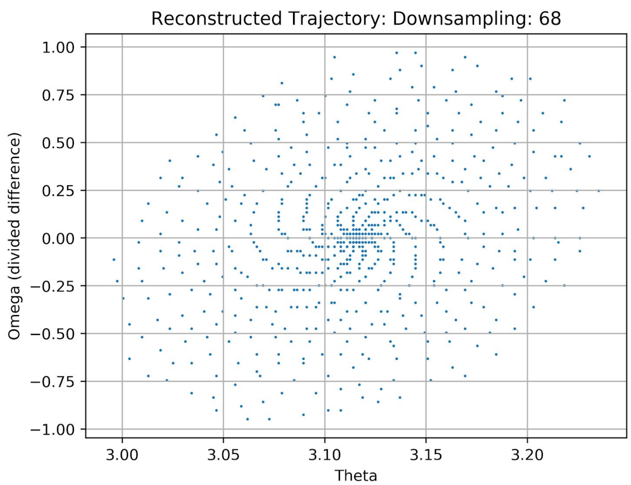
CSCI 4446 Spring 2019 PS8
Derek Wright

1.







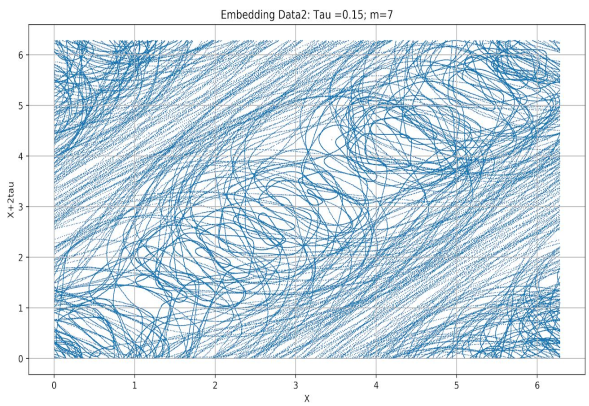
The plots shown above do **not** look like a clean spiral, as the trajectory should. Instead we see a striated response that is quite dependent on the subsampling frequency. At smaller downsamplings (the term I use to describe how many datapoints I skip before taking another point), we see a horizontally stratified response, while at high downsampling frequencies we see several arms of a spiral instead of the one cohesive trajectory we should see.

Because we are constructing this trajectory by taking a numerical derivative of our theta data (heh, that's fun to say), we are not only getting inherently erroneous data (we don't have an infinite time step like a real derivative does). Not only that, but derivatives by their nature amplify noise. So if there is any error in recording the theta data, this error will be magnified during a reconstruction such as this. And of course, since this data was taken with a real-life encoder, there will be noise present. I believe this is the cause of the erroneous trajectory plotted above.

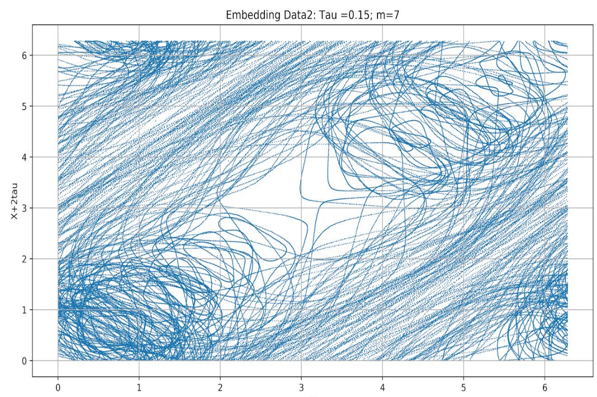
2. a

Below I show each 250 second data2 file separately, and then all together. You will notice that no one looks the same, and that the final one is a more-or-less filled in space from 0 to 2pi on both axes. This would tell me that this trajectory represents a **chaotic attractor**, since it appears as though no cycle repeats itself, and the trajectory is very dense (note that data2 goes for 1000 seconds, which is a long time and enough for any transient to die down. Even still, we see no limit cycle in the later plots of 250 seconds.).

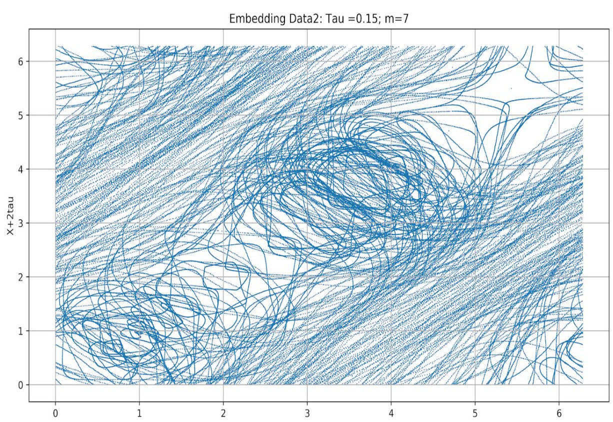
First 250:



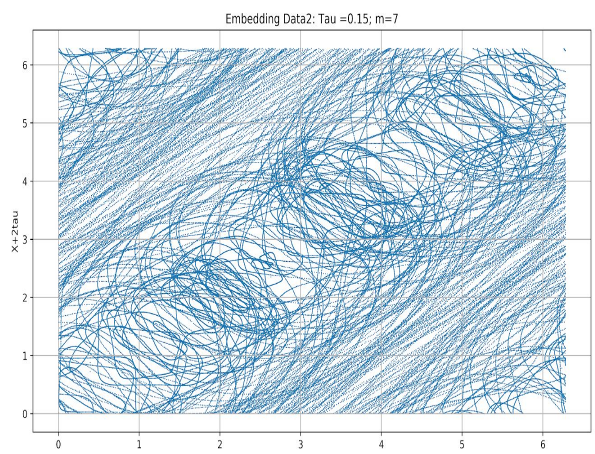
Second 250:

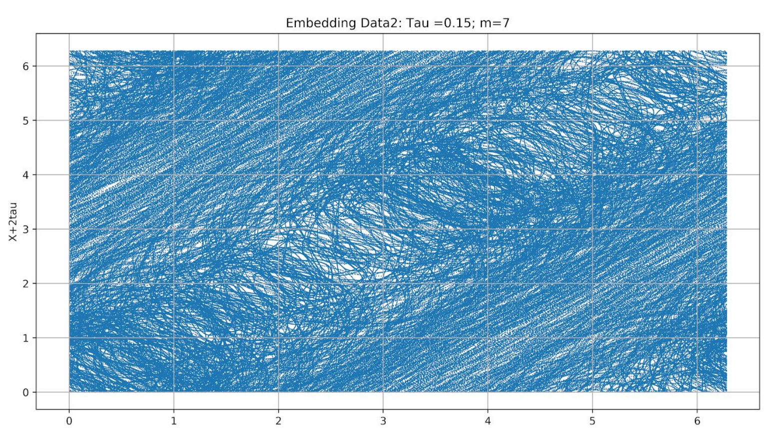


Third 250:



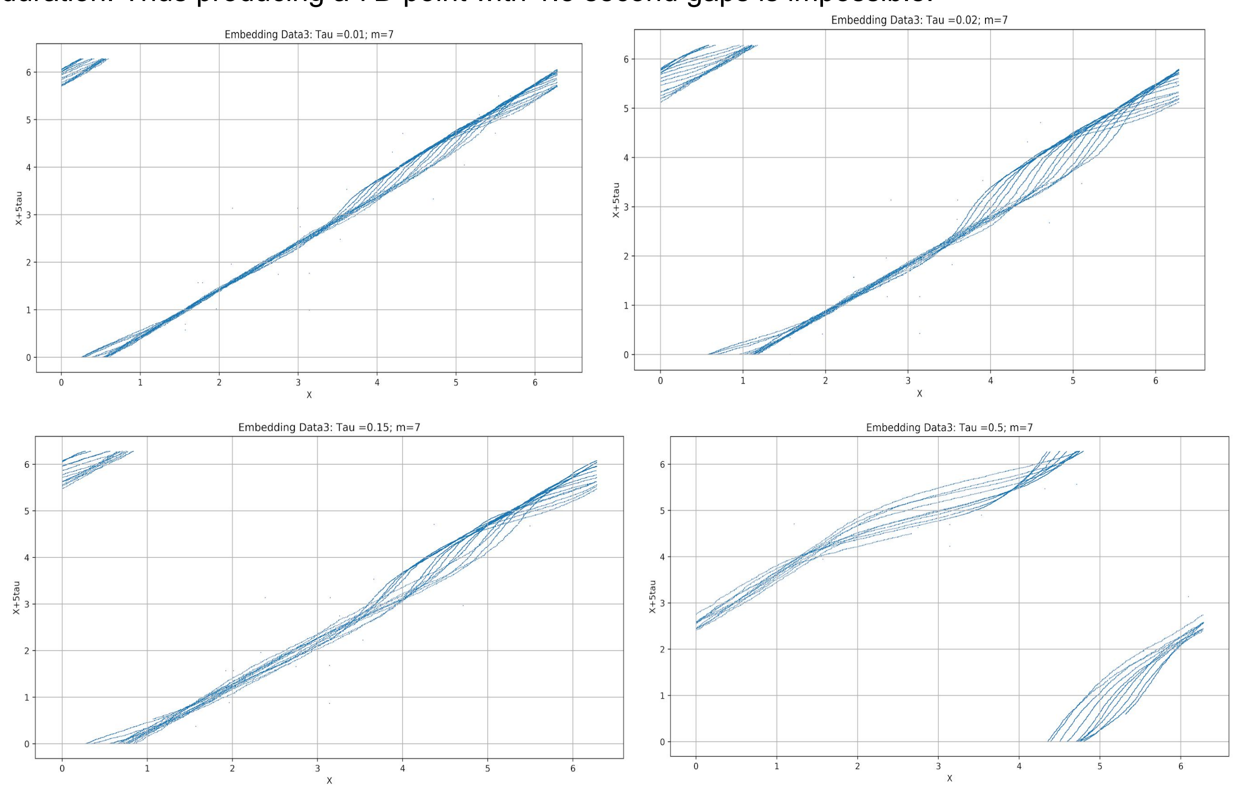
Fourth 250:





b.

Below are some plots of θ(t) vs θ(t+5τ) are shown below. Expectedly, the plot begins near the diagonal and expands off of it for larger τ (as is seen in the τ = 0.5 case -- bottom right below). The structure mostly stays the same though. That is, I would say this trajectory is a limit cycle. The small striations/deviations from that cycle are attributable to the noise in all dimensions projected down onto this 2-dimensional representation. Interestingly, I used mutual on this data set and found that a τ = 20* Δt = 0.02 was the first minimum. I have included that plot to show what *should* be the best representation of this trajectory.
Also, I was unable to reach 1.5 seconds for τ because the data3 dataset is only 10 seconds in duration. Thus producing a 7D point with 1.5 second gaps is impossible.



3. a

Takens' Theorem tells us that for system with *m* state variable, we may need as many as 2m+1 reconstruction dimensions. For the case of the driven pendulum, we have the usual state variables θ and ω. But we also have the applied torque on the driver motor, which changes over time as well, beholden to the drive amplitude and frequency. This makes the driven pendulum a 3D system, and therefore we would need 7 dimension in reconstruction space to guarantee we have enough dimensions. Of course, chaos is only possible with 3 dimensions, so whatever model you want to use to introduce a third state variable (motor current, torque, force, etc) is valid.

For the undriven pendulum, the system is only 2D, so we would only need 5 reconstruction dimensions in that case.

b.

If we change m to 2, I believe that the trajectory would cross over itself many times, which means we have lost information about that trajectory, since we would need more dimensions to see them pass under or over each other.

If m=25, the reconstructed trajectory would fill out the full range of theta in all 25 axes, thus causing a very dense trajectory that is not very useful or informative, since the information is being overshadowed by superfluous dimension, that each need a full range of theta points plotted

c.

If τ were very small, and the sampling frequency is much faster than the dynamics, many consecutive theta points would be the same value (or very close), thus causing a point in reconstructed space right on the diagonal. This results in a diagonal line, since the data has a correlation coefficient very close to 1.

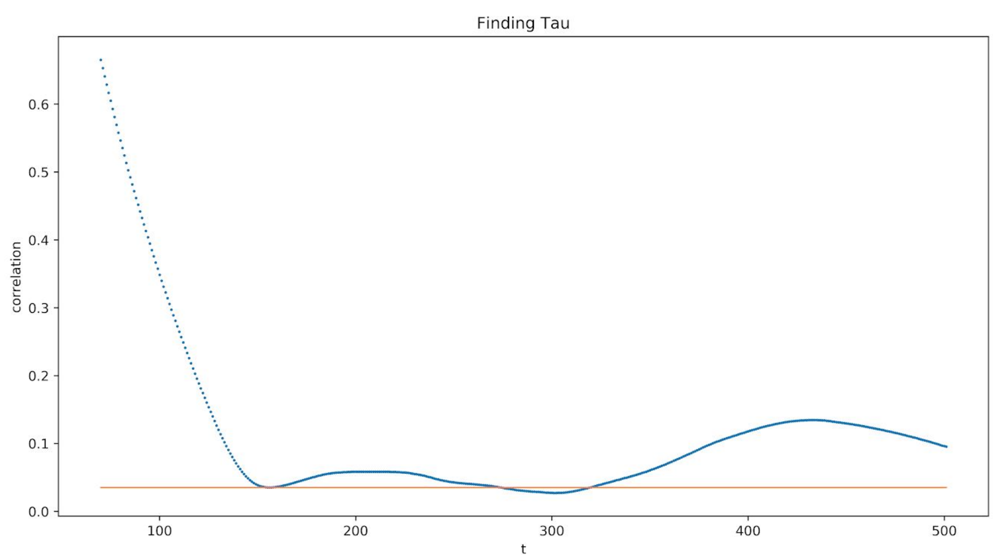
If τ were very large, we would firstly need a **lot** more data in order to reconstruct 7-dimensional points. Assuming we did have enough data, though, we would see that the trajectory fills up our space, since the correlation of the data is very close to zero.

4.

Using the mutual function like so:

./mutual -D500 ../../ps8data/data2.first250sec -o ../../TiseanOuts/data2_f250.txt

I was able to produce the below plot. I then found the first local minimum (using argrelextrema) to be 156 * Δt.



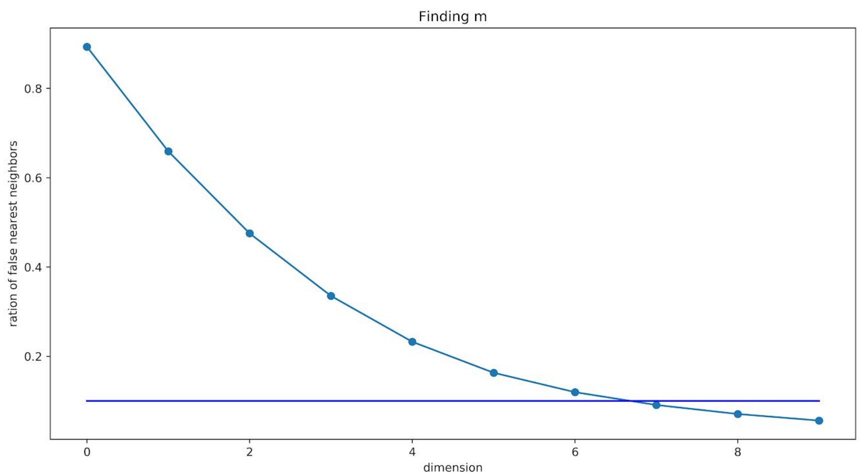
Data2 uses a Δt of 0.002. This means τ = .312.

5.

Using τ = 156, and calling the false nearest function:

./false_nearest -M1,10 -d156 ../../ps8data/data2.first250sec -o ../../TiseanOuts/fn_data2_f250.txt

I was able to produce the below graph, with a horizontal line at 10% false neighbors



As can be clearly seen, m=7 is the first dimension that we see less than 10% false neighbors. This further give us confidence that 7 is the correct number of dimensions to use for this system.