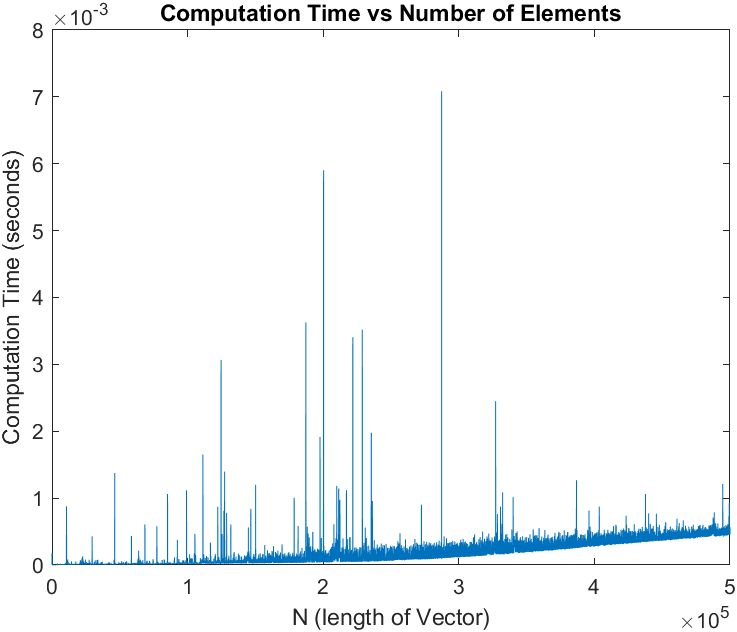
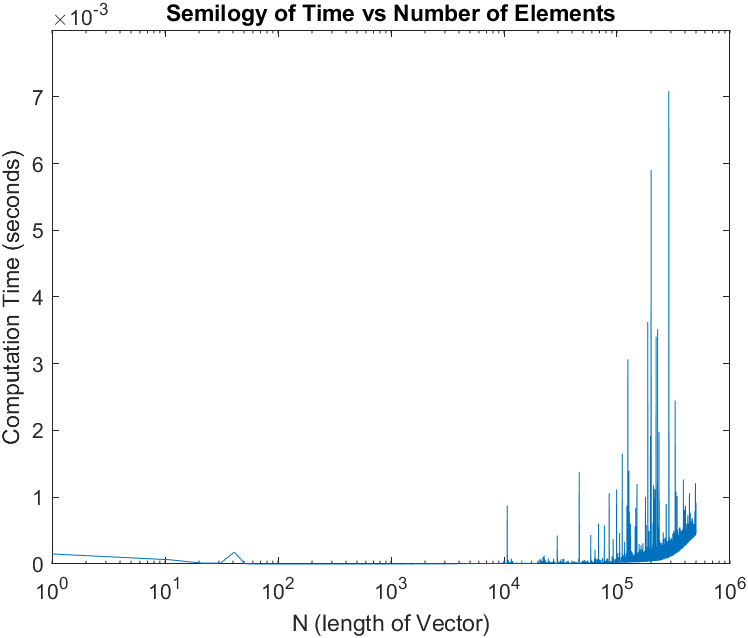
# Intro to MatLab

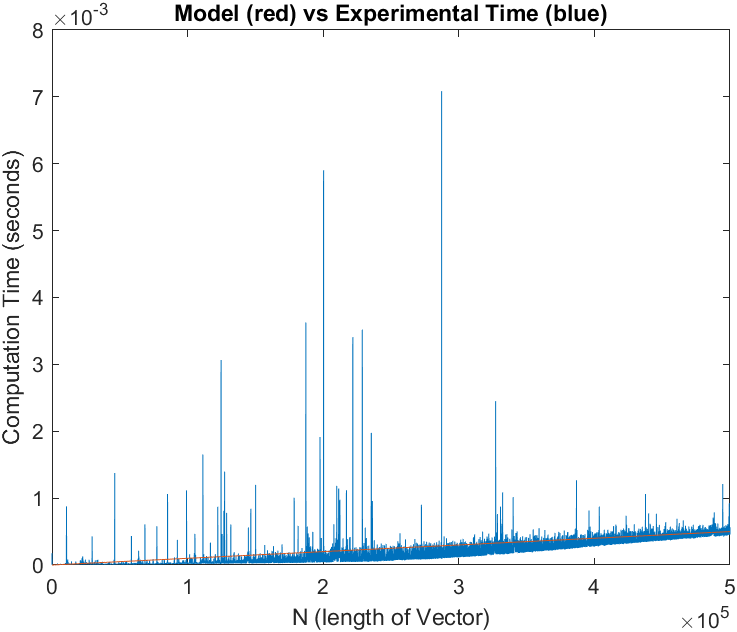
In order to become familiar with Matlab, I have been assigned several experiments that will explore the computational complexity of several computational functions for vectors and square matrices. Source code is provided separately.

## Experiment I – Random Vector Dot Product Complexity

This experiment measures the time taken for the dot product of two random valued, N-length vectors. For my purposes N ran from 1 to 500,000 in steps of 10. The entire data collection process took 220 seconds on a 2nd generation i5 processor. The results are plotted below conventionally and using semilogy.

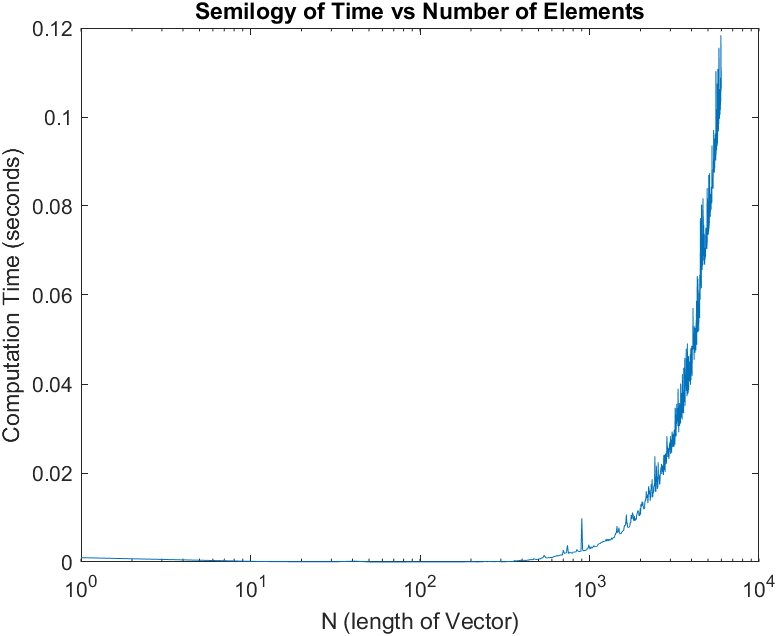
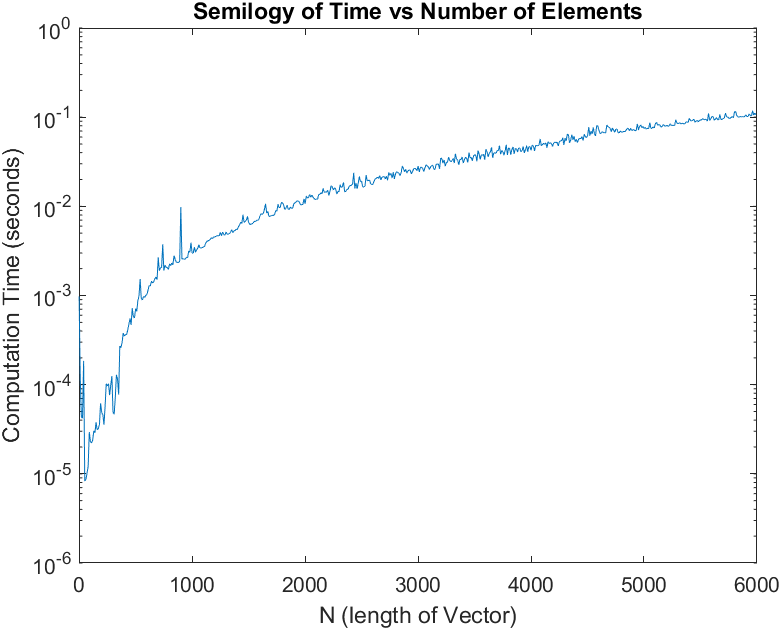


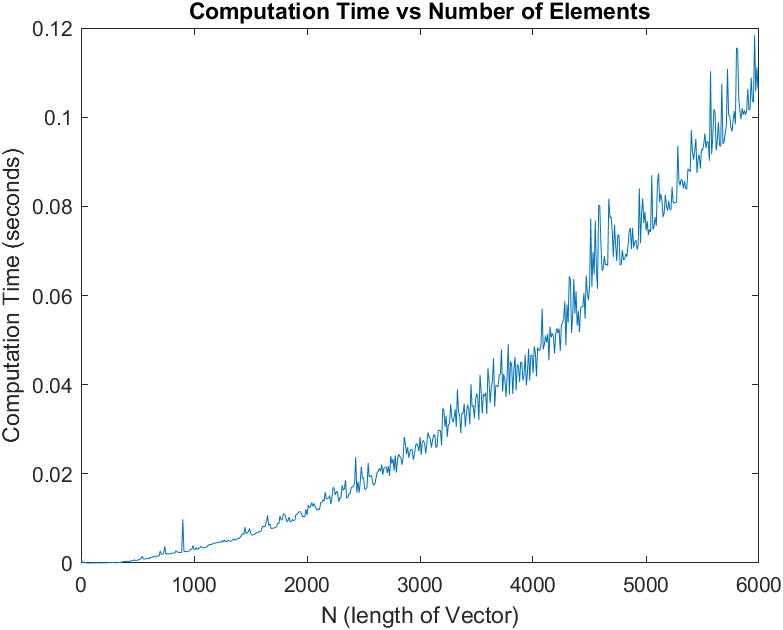
The computation time appears to increases linearly, *O(N).* There are a few outliers, I expect this is due to other processes running on my machine during timing. Creating a predictive model yielded :

****

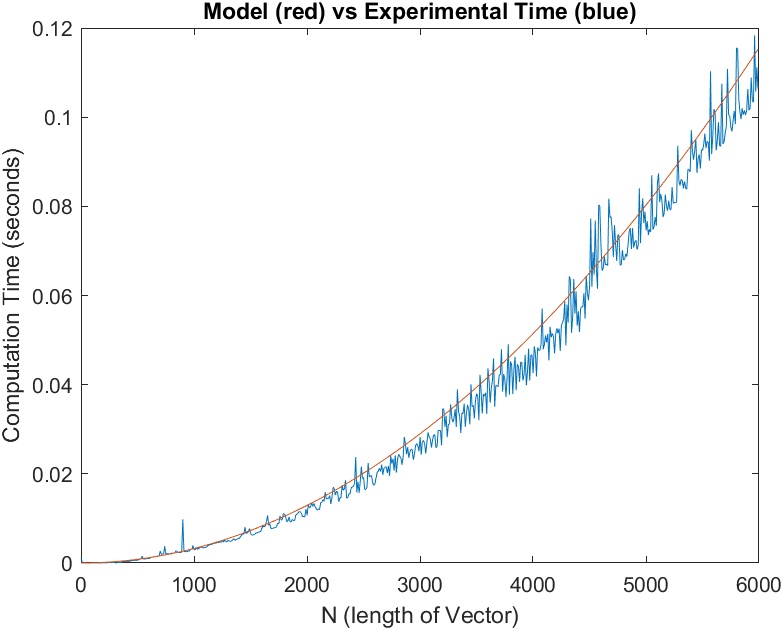
## Experiment II – Random Square Matrix Dot Product Complexity

The second experiment measures the amount of time taken for a dot product of two random valued square matrices of N x N dimensions. For this experiment N runs from 1 to 6,000 in steps of 10. The entire measurement process took 154 seconds on my 2nd generation i5 processor.



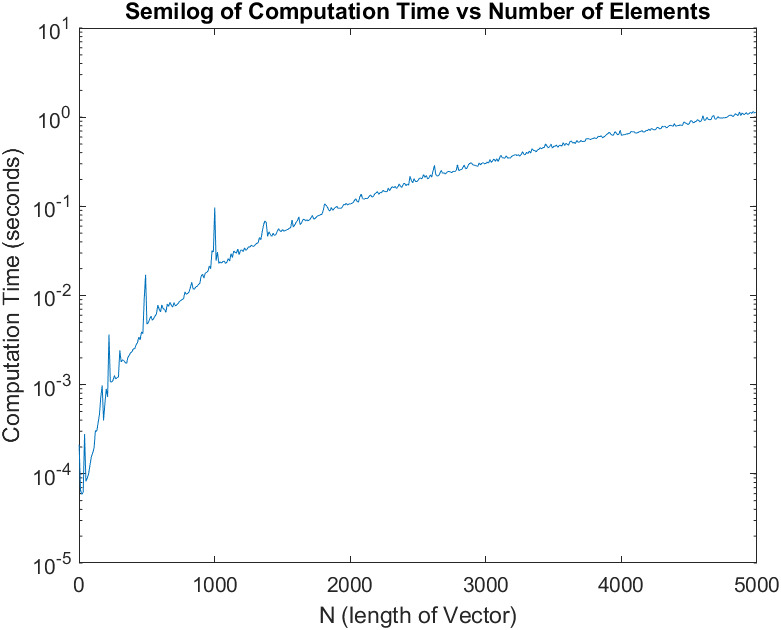
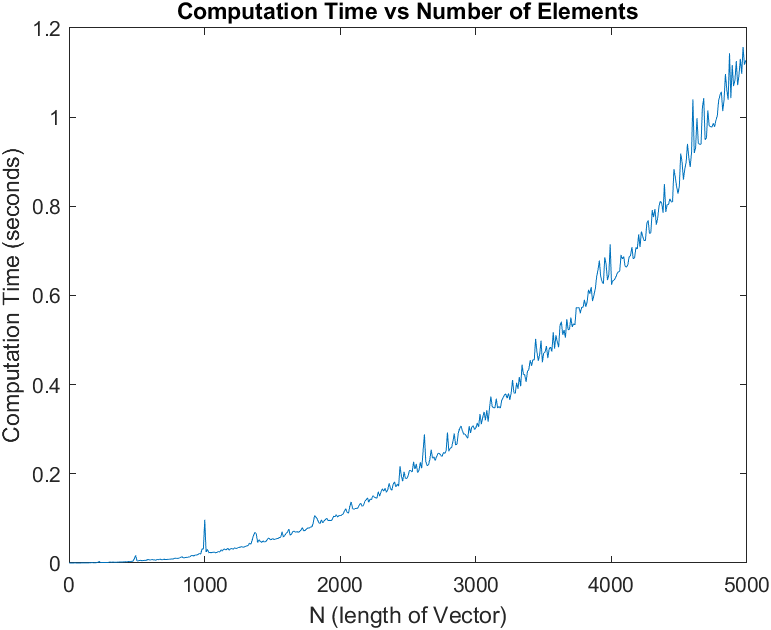


As expected, this appears to be a harder computation. Looking to be in the neighborhood of complexity. Creating a predictive model yielded:

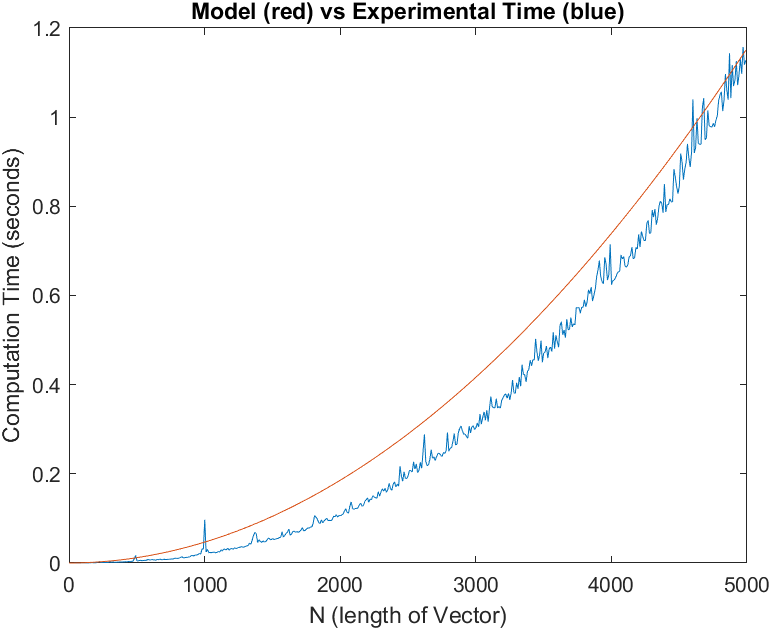
**

## Experiment III – Random Square Matrix Solution Complexity

The second experiment measures the amount of time taken to solve a linear system of random valued N x N dimensions and a random valued solution matrix of N x 1 dimensions. For this experiment N runs from 1 to 5,000 in steps of 10. The entire measurement process took 244 seconds on my 2nd generation i5 processor.

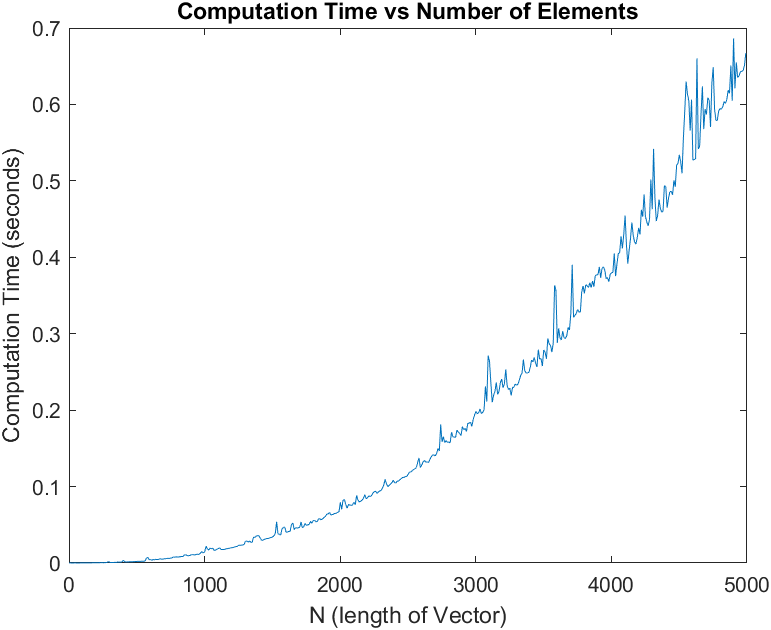
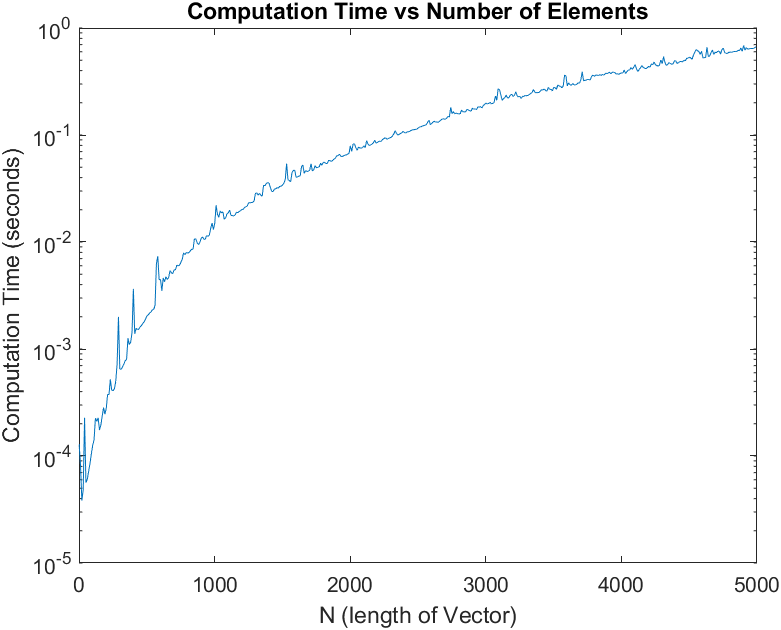


The runtime appears to be , this needs a slightly different model because it’s a little slower on my machine. The model yields the following results:



## Experiment IV – Contrived Square Matrix Dot Product Complexity

The second experiment measures the amount of time taken to solve a linear system of N x N dimensions with the primary diagonal being 1 and the first super diagonal to either side of the primary diagonal being -.5 with a random valued solution matrix of N x 1 dimensions. For this experiment N runs from 1 to 5,000 in steps of 10. The entire measurement process took 134 seconds on my 2nd generation i5 processor.

The runtime appears to be , modeled as:

