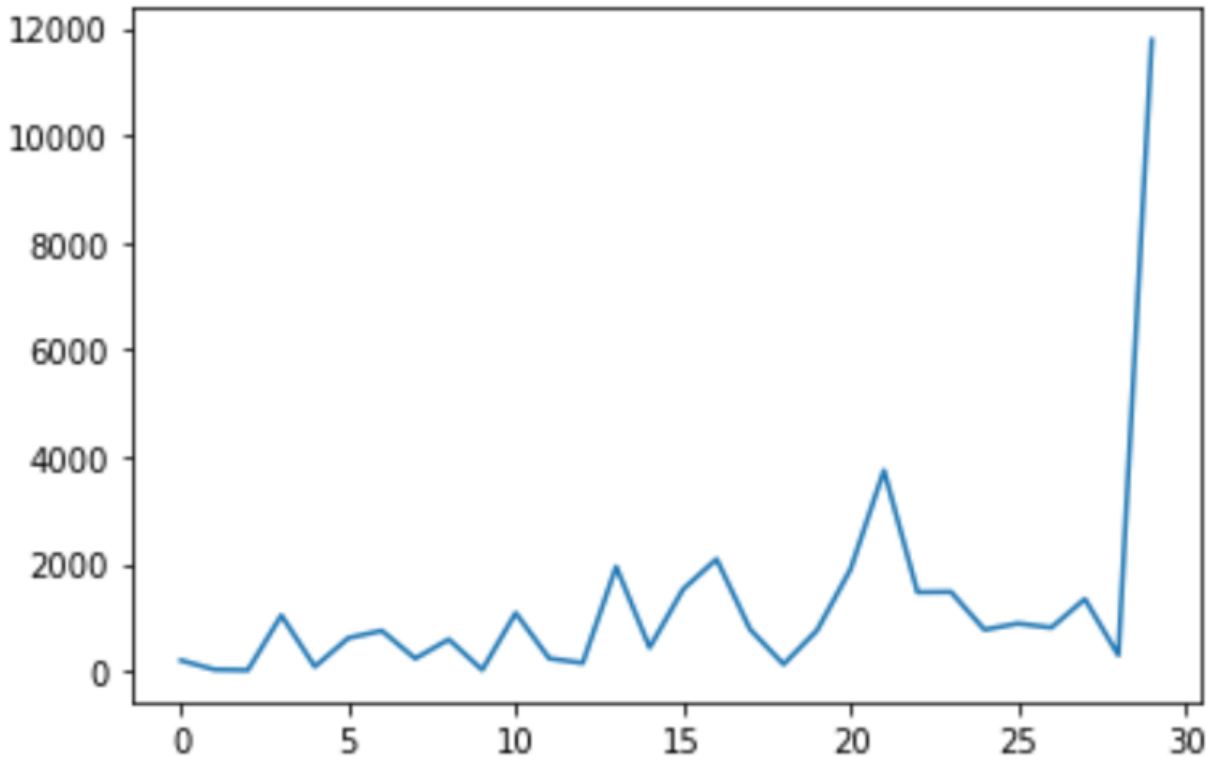
Michael Ryvin

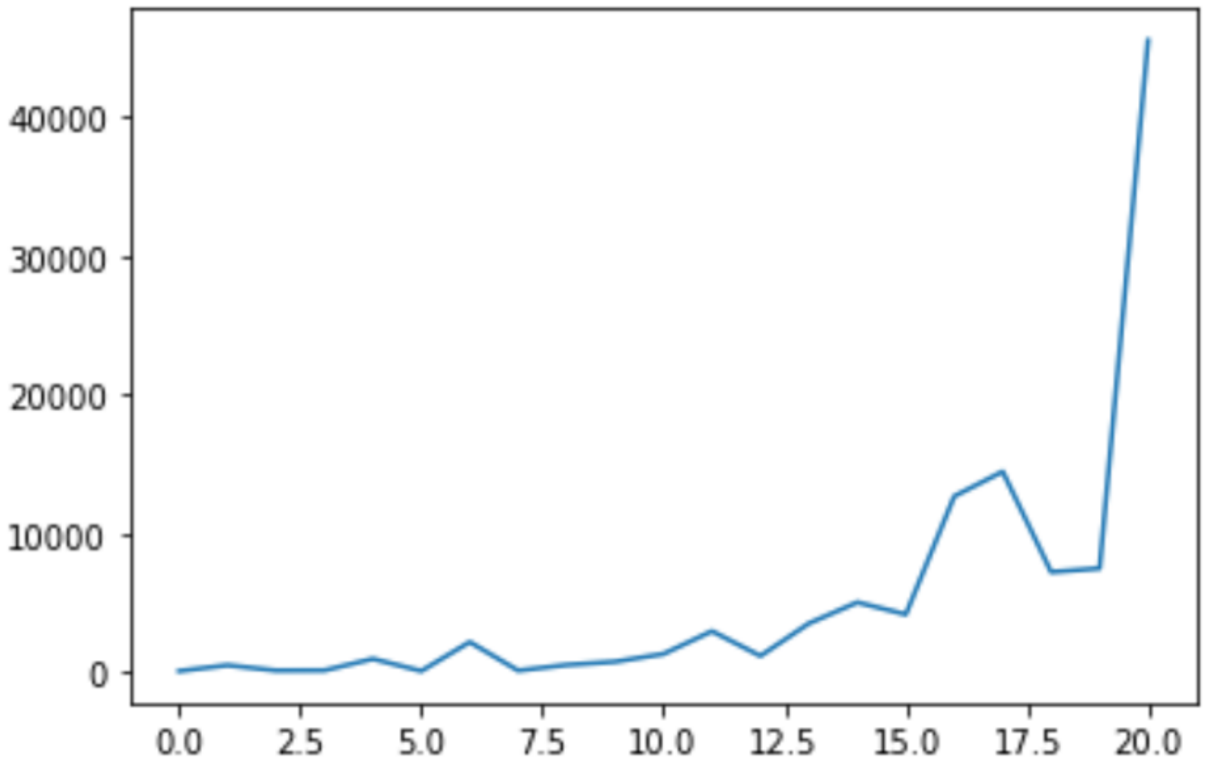
**Problem 1:**

Plot of k values 1-30:




Based on when the graph curves, this suggests that the data has a dimensionality of somewhere between 25 and 30.

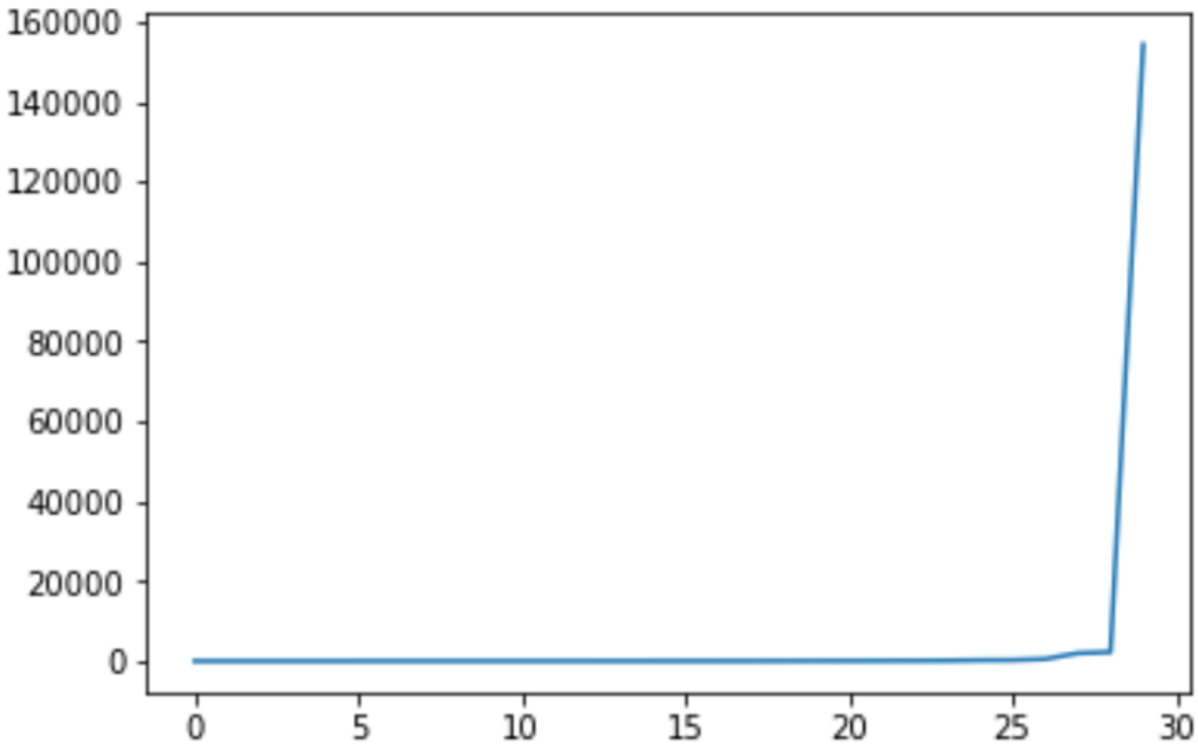
Plot depending on values of sd:




The values of the x-axis are correct, but the labels are incorrect. 2.5 refers to an sd of 0.25, 10 refers to an sd of 1, and so on

The error increases as the sd increases. This is because a higher sd introduces more randomness in the data. By introducing more randomness, the data becomes harder to predict, and so it becomes more difficult to make a model to predict it.

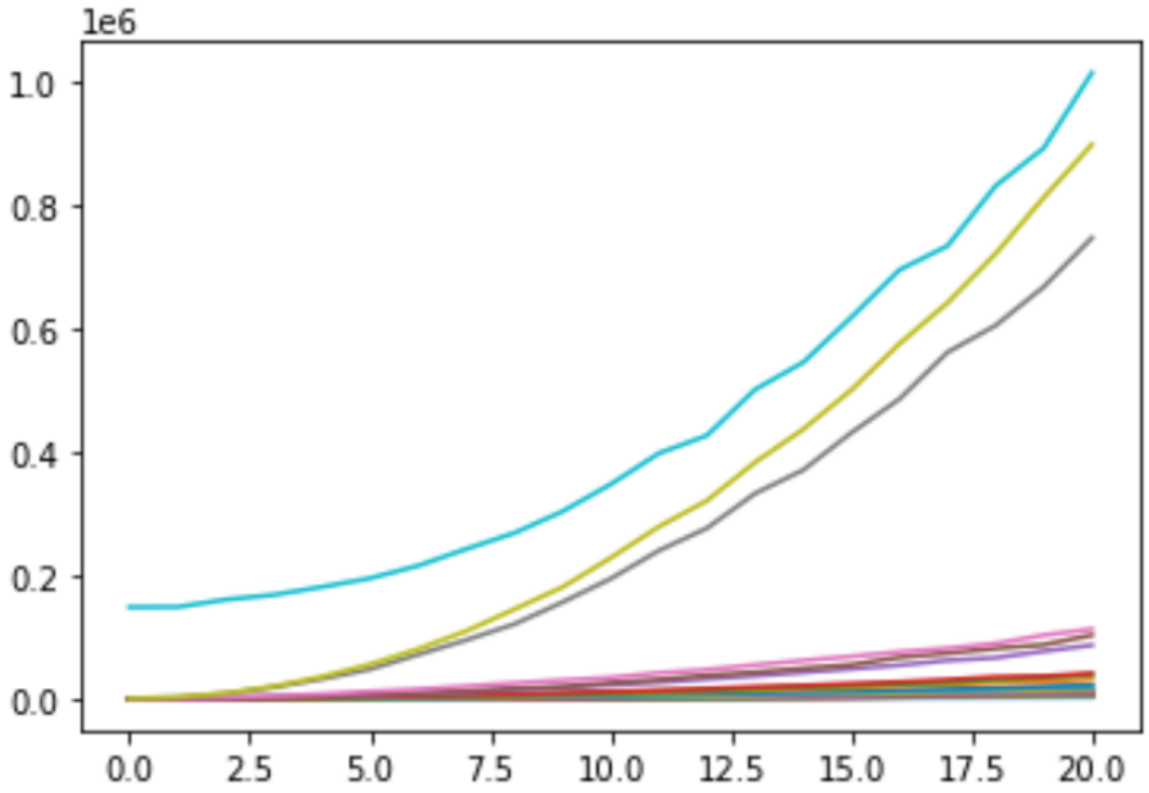
**Problem 2:**



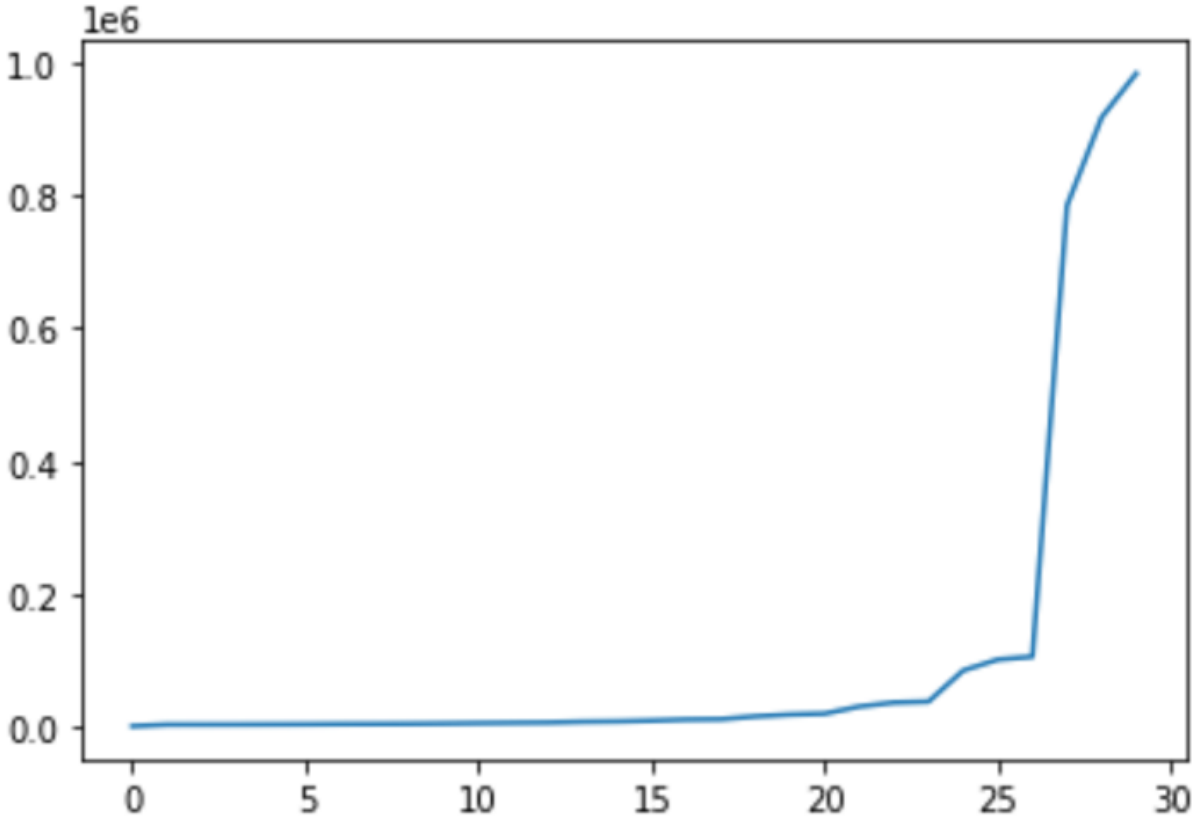

```
[1.08343226e+01 1.21460981e+01 1.25326714e+01 1.27968358e+01
 1.35275482e+01 1.37639244e+01 1.43390888e+01 1.50545040e+01
 1.60456515e+01 1.66729625e+01 1.83694654e+01 1.94911203e+01
 2.10763655e+01 2.34071375e+01 2.53383336e+01 2.83242955e+01
 3.29454441e+01 3.57327738e+01 4.45633377e+01 5.04757782e+01
 6.16606163e+01 7.95782670e+01 9.26105885e+01 1.26483845e+02
 2.21760286e+02 2.61651403e+02 5.05630235e+02 1.97168251e+03
 2.26805095e+03 1.46382475e+05]
```

The eigenvalues suggest that the dimensionality of the data is somewhere between 25 and 30.

The data is very robust, as using different datasets results in essentially the same results.




As the sd increases, the value of the eigenvalues increases. The graph above shows all of the eigenvalues plotted depending on the sd.




The graph above has an sd of 2. You can see more clearly that, when compared to the graph with an sd of 0.1, the eigenvalues are much larger on the y-axis. Also, while the dimensionality is still between 25 and 30, it appears that it may be slightly closer to 25 than in the example with 0.1 for sd.

**Problem 3**

For this problem, I did not know how to perform the linear regression as I had not been able to do so in previous assignments. As such, I answered the questions by using a ready-made linear regression algorithm.

I could not figure out how to make a dependency graph, but the following were the top five predictive variables for each variable

X1

- X2, X4,, X3, X14, X17

X2

- X1, X26, X5, X15, X14

X3

- X1, X6, X11, X25, X14

X4

- X7, X1, X10, X22, X5

X5

- X2, X8, X27, X19, X23

X6

- X9, X3, X25, X27, X10

X7

- X10, X4, X1, X6, X13

X8

- X5, X11, X19, X22, X6

X9

- X6, X17, X19, X12, X20

X10

- X19, X7, X13, X6, X3

X11

- X14, X8, X3, X5, X6

X12

- X22, X17, X15, X9, X19

X13

- X16, X15, X10, X5, X9

X14

- X1, X17, X11, X20, X3

X15

- X22, X18, X12, X25, X1

X16

- X19, X13, X5, X25, X28

X17

- X20, X14, X12, X9, X30

X18

- X21, X15, X24, X26, X29

X19

- X25, X22, X16, X8, X5

X20

- X17, X14, X12, X23, X26

X21

- X18, X24, X26, X23, X29

X22

- X30, X12, X19, X25, X15

X23

- X5, X26, X20, X21, X8

X24

- X21, X27, X26, X18, X5

X25

- X19, X22, X28, X16, X6

X26

- X21, X29, X2, X23, X24

X27

- X30, X5, X24, X22, X6

X28

- X25, X19, X16, X3, X18

X29

- X26, X19, X18, X21, X14

X30

- X22, X20, X27, X17, X25

If I were to create a dependency graph connecting the two most important variables, the result would NOT be robust. When I repeat the code over and over again, the two most important variables changes. While some variables appear more often than others for each variable, it does change.

- For example, for X1, in one attempt the two most important variables were X3 and X4, while in another attempt the two most important variables were X2 and X3

This same thing would apply whether the graph was made using the two most important variables, three most important variables, four most important variables, or so on.

It is difficult to see exactly how a higher sd affects the results

- However, it seems that a higher sd results in a higher likelihood that the most predictive variable is X1 for the other variables
- The most predictive variables tend to always be the variables closer to the variable, though many exceptions exist
    - Variables close but away by a multiple of 3 tend to be fairly predictive as well regardless of sd