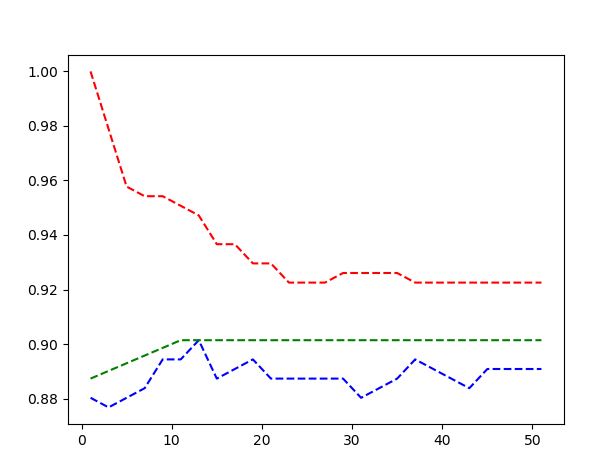
**Part 1 KNN**

****

In figure 1, where accuracy is plotted as a function of K, red is training error, blue is test error, and green is testing error of the leave one out cross validation method.

The training error, shown in figure 1, starts out at 100%. Since k equals 1, it is choosing itself as the nearest neighbor and thus is not a good indicator of training error. It can be seen converging at an accuracy of 93% around k = 23. Testing error, shown in figure 1, remains relatively stable around 89%, though it does seem to increase slightly as the training error drops. The cross-validation testing error has a similar, but higher trend in accuracy which converges at 90% after k = 11.

Since the cross-validation error doesn’t change after k = 11, it is a good assumption that having a higher k would not be useful.

**Part 2, Section 1:**

**Information gain : .109542253521**

**Training error rate: .0598591549296**

**Testing error rate: 0.105633802817**

**Thoughts on these values: When I implemented this functionality, I tested using both the data given when normalizing the values, and when not. In both cases the results were almost the exact same, so I left normalization out.**