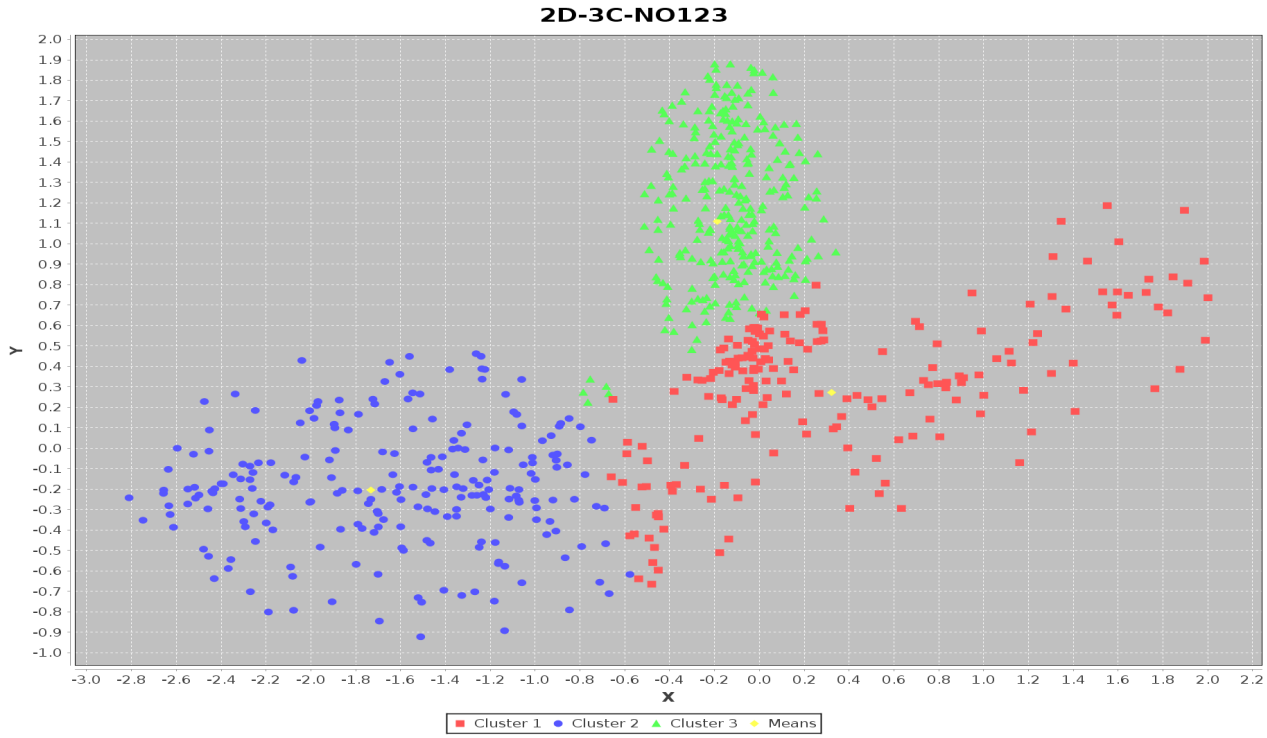
Michael Gillan

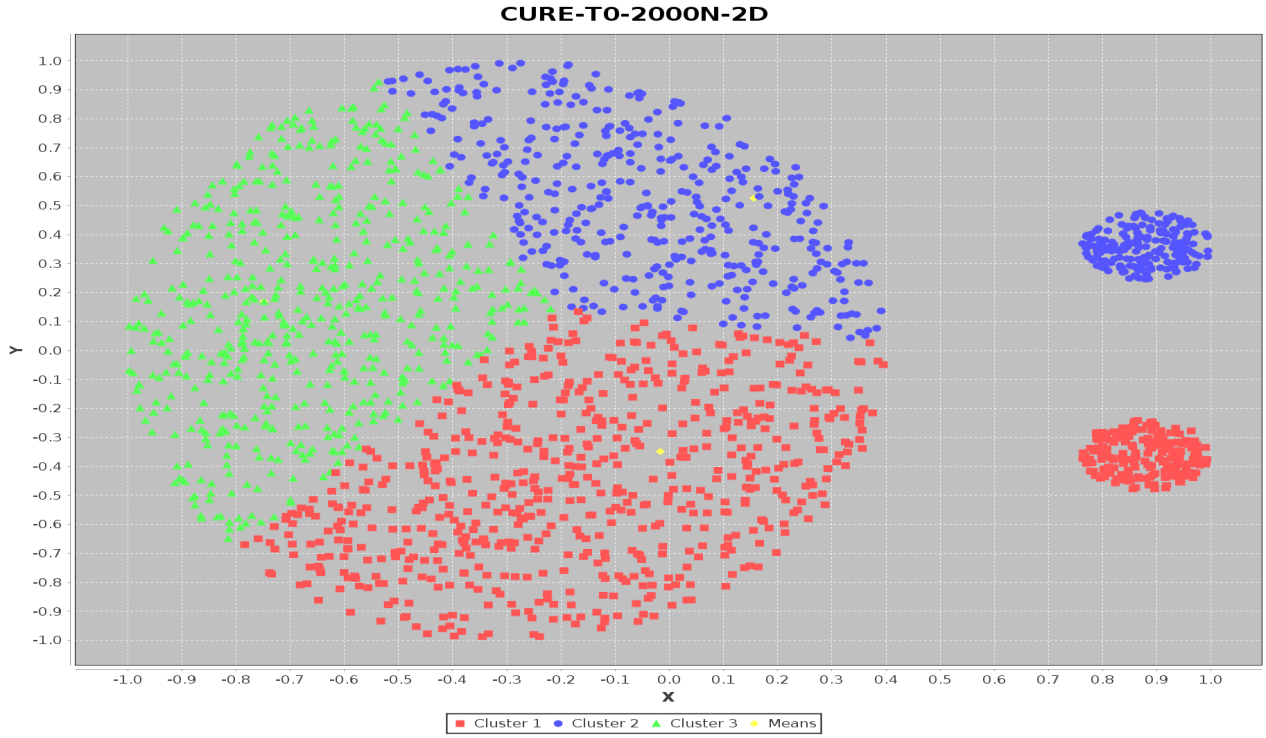
K-Means

First dataset:



This is the first data set I tried the clustering algorithm with. The initial data appears to be in 3 seperate clusters, and so the is the k I decided to go with. You can see that the centroids appear to have more or less converged to each of the 3 clusters. However there is some overlap between the two clusters on the left.

Second dataset:



I chose this dataset as I think it nicely shows the shortcomings of k-mean. Here, there are 3 distinct clusters, 2 dense and 1 sparser. However, k-mean with a k value of 3 ends up dividing the larger cluster instead of seperating the 3 clusters. This is partially due to the way I select the initial centroids: the Forgy method uses existing data points as the intial centroids, which in this instance means they are more likely to fall inside the larger cluster. It is also due to k-means tendency to create equi-sized clusters.

Pros and Cons of k-mean

By taking the k as an input parameter into the algorithm, k-means can perform porrly if a bad choice of k is made. It can also tend to a local minimum, which is not the generally expected result, as shown in the second dataset above.

However, these drawbacks also mean that k-means is very efficent, and can run on large data sets quickly.