Instruction

The program is an implementation of k-mean clustering algorithm.

K-means clustering is a method of vector quantization, originally from signal processing, that is popular for cluster analysis in data mining. K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest distance.

The algorithm runs like this:

1. Initially select k centroids of clusters randomly. In my program, I just randomly chose k elements as the initial centroids.
2. Assign elements to it's nearest centroids. In my program, there are two distance measurement functions. Default measurement is L2 distance.
3. Calculate new centroid for each cluster.
4. Repeat step 2 and step 3 until the centroid is not changing.

Example:

The test sample is 2D pointers.

Originally they distributed like this:

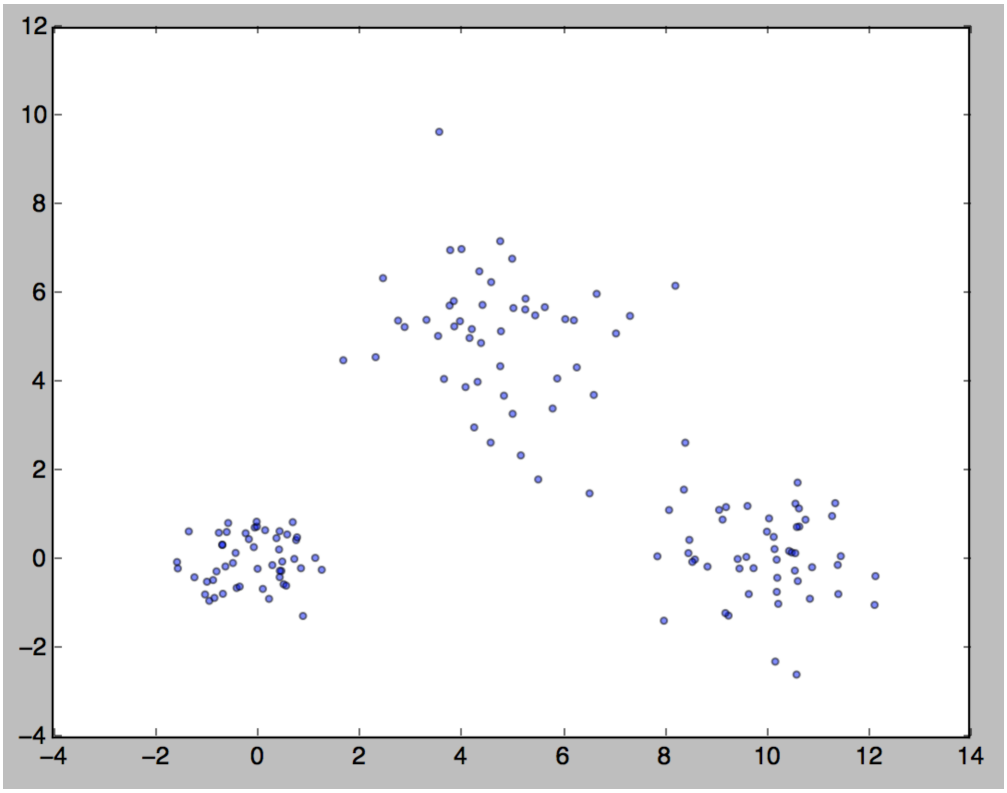

Figure 1 Original points

After several iterations, we got the clustered result. The red points in the following figure are the final centroids.

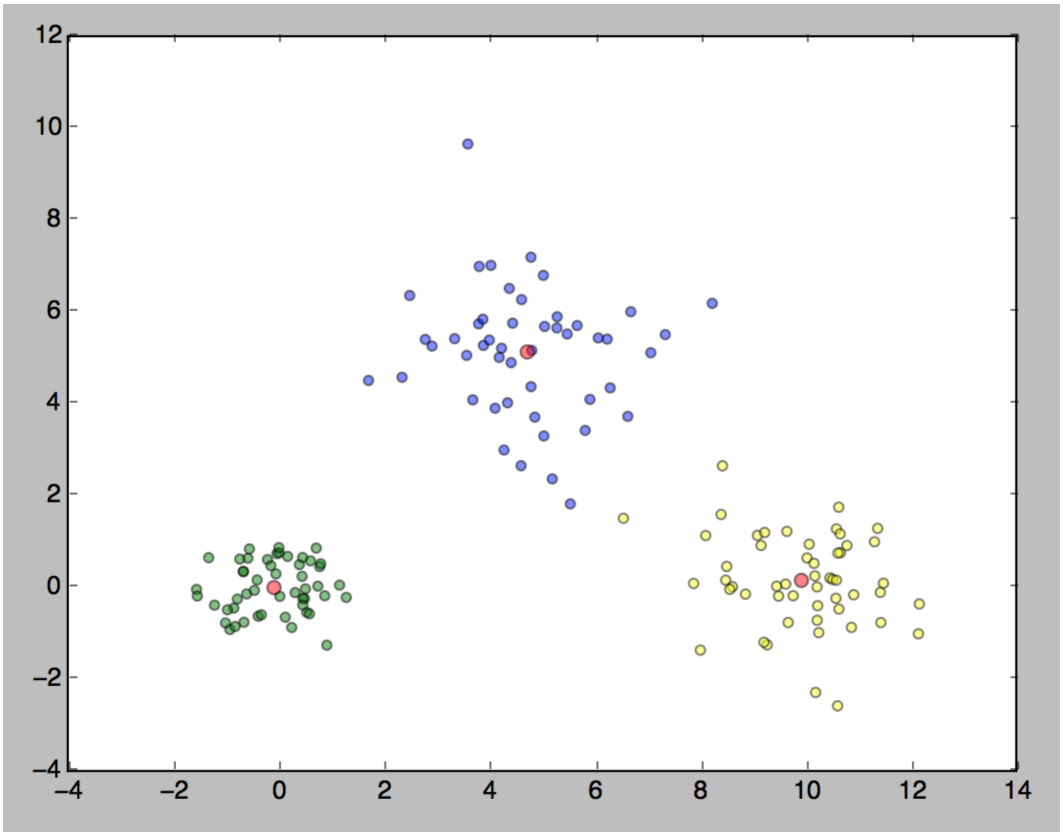

Figure 2 Clustered points

The program was written in Python3. Multi dimensions elements are supported. But the visualization version requires matplotlib module and only support 2-Dimension elements, i.e. 2D pointes. The code is self-explanatory.