

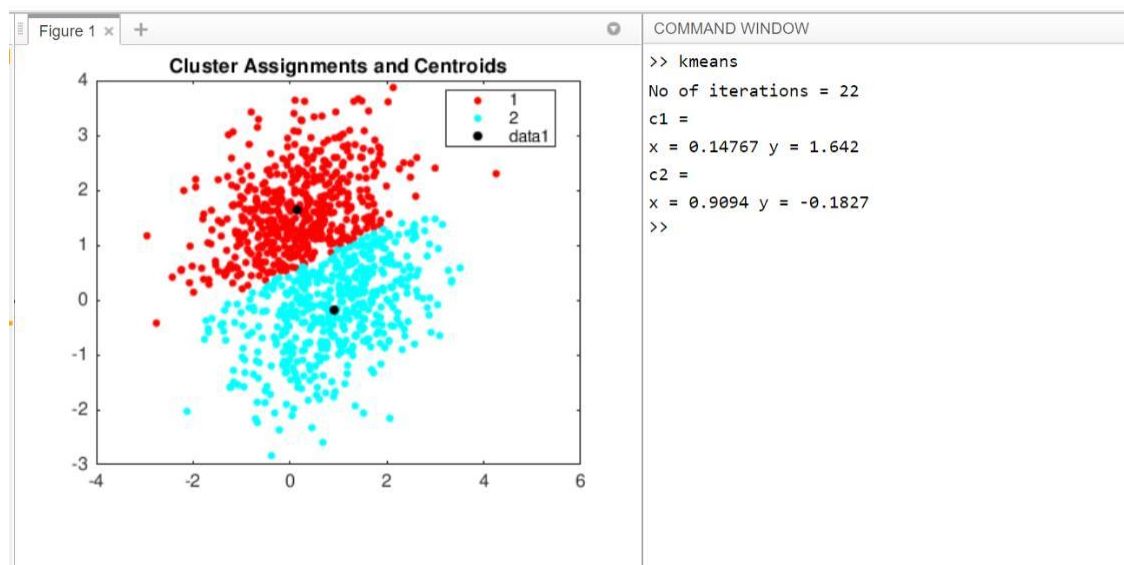
Implementation:

1. Initialize input data
2. Create 2-D Gaussian random data using `builddataset(mu,sigma,n)`
3. calculate clusters and centroid using `mykmeans(data,k,c)`
4. `mykmeans` evaluates new centroid till the l2-norm of old and new centroid is ≤ 0.001 . This implementation calls out other function which performs different steps of kmeans
 - i. `caldistances(X, k, c)` - calculates distance between each data point of X with all present centroids in c
 - ii. `assigncluster(dist)` - Assigns cluster to data point in X based on its distance from each center point in c by comparing distances saved in dist matrix.
 - iii. `calnewcentroid(newclusters,data)` - calculates new centroid with respect to each cluster and data points present in that cluster
 - iv. repeat all three steps till `l2norm` returns false. `checkl2norm(newcentroid,k,c)` – returns false when l2-norm of all old and new centroids ≤ 0.001
 - v. return cluster with all data points in X and the clusters they are assigned to.
5. prints total no of iterations, final centroids and scatter plot using `display(cluster, centroids)`

Report :

Input : $k = 2$, $c1 = (10; 10)$, $c2 = (-10;-10)$

Output : **Note - data1 denotes centroid**



Input : $k = 4$, $c1 = (10; 10)$, $c2 = (-10;-10)$, $c3 = (10;-10)$ and $c4 = (-10; 10)$

Output : **Note - data1 denotes centroid**

