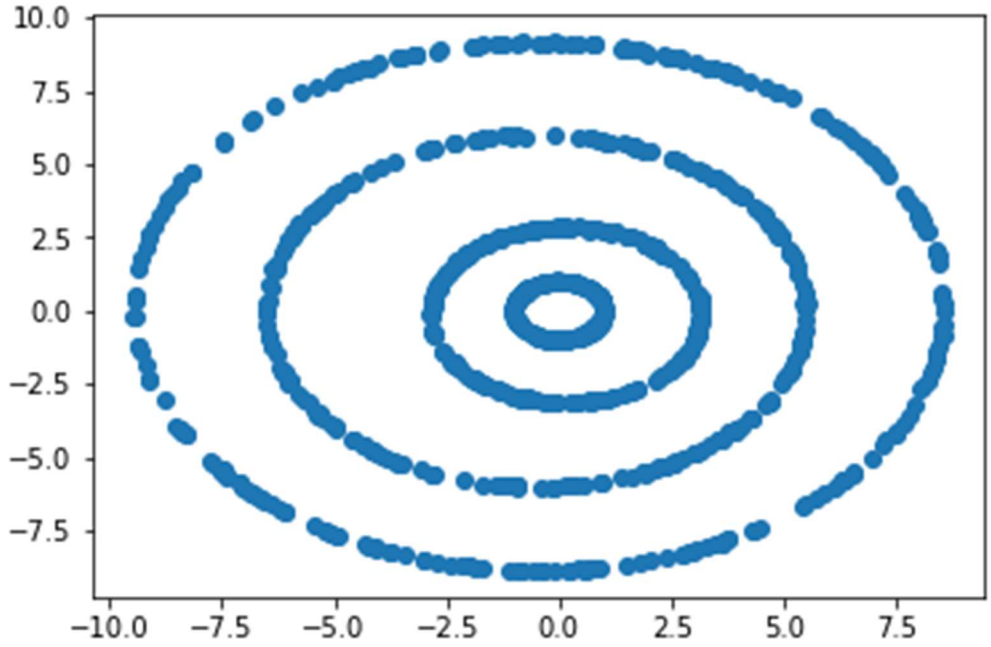
Question 1

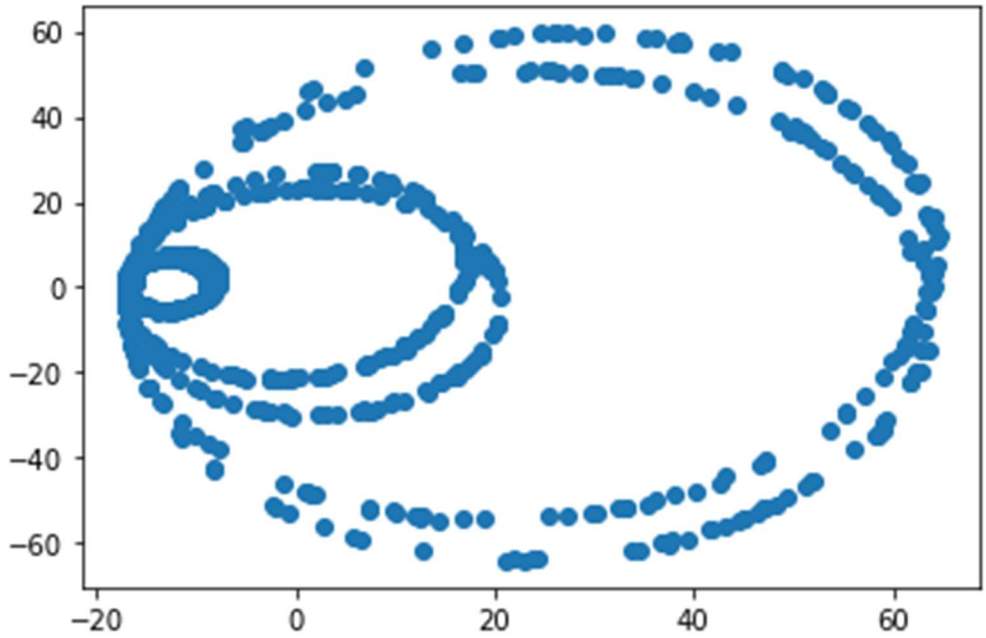


1.varience along PC1 is 54.178024528852234
varience along PC2 is 45.821975471147766

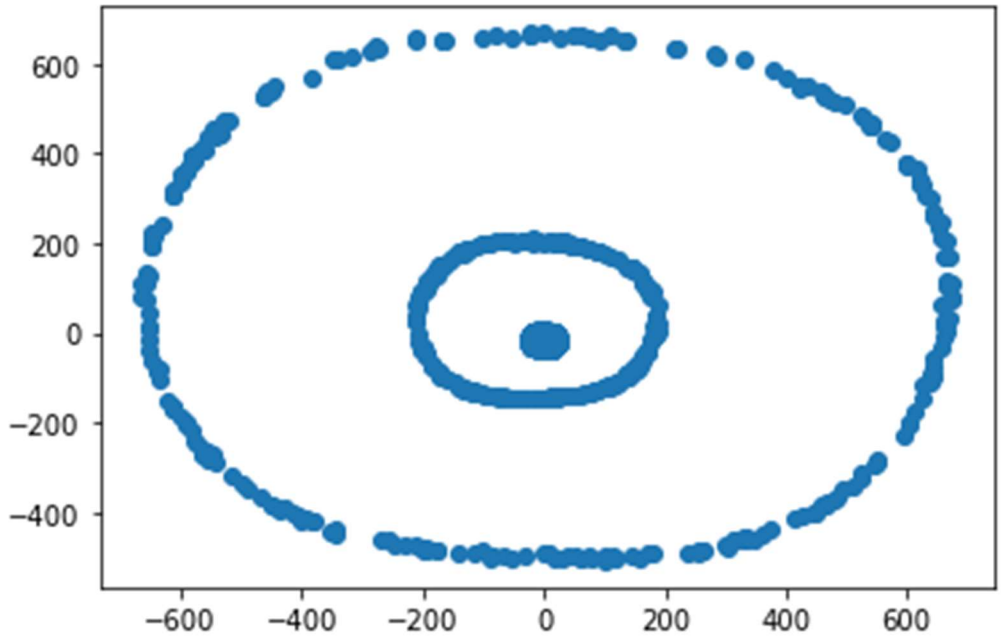
2. After applying centering, my data is same and hence the variance Anong PC1 and PC2 is also same. Hence we can see here that centering does not help

3(A). **κ(x, y) = (1 + x T y) d for d = {2, 3}**

For d=2

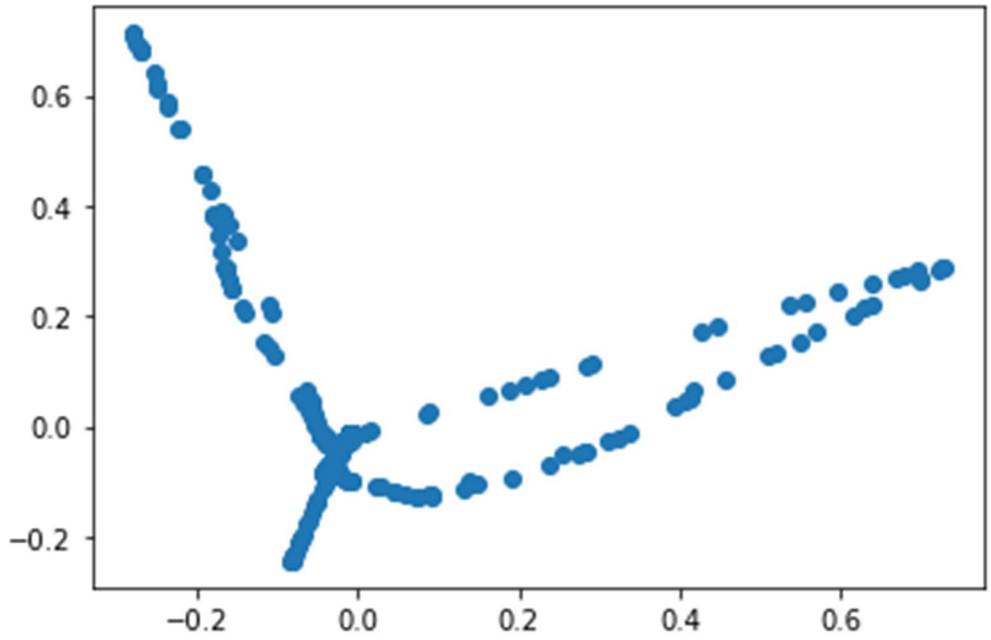


For d=3

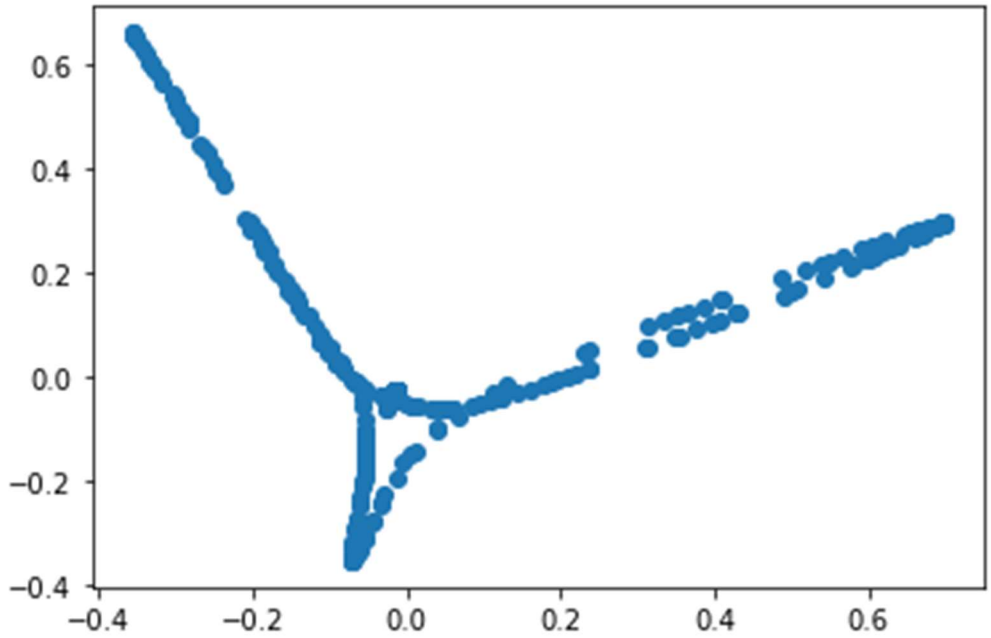


B. κ(x, y) = exp −(x−y) T (x−y) 2σ2 for σ = {0.1, 0.2 , . . . , 1}

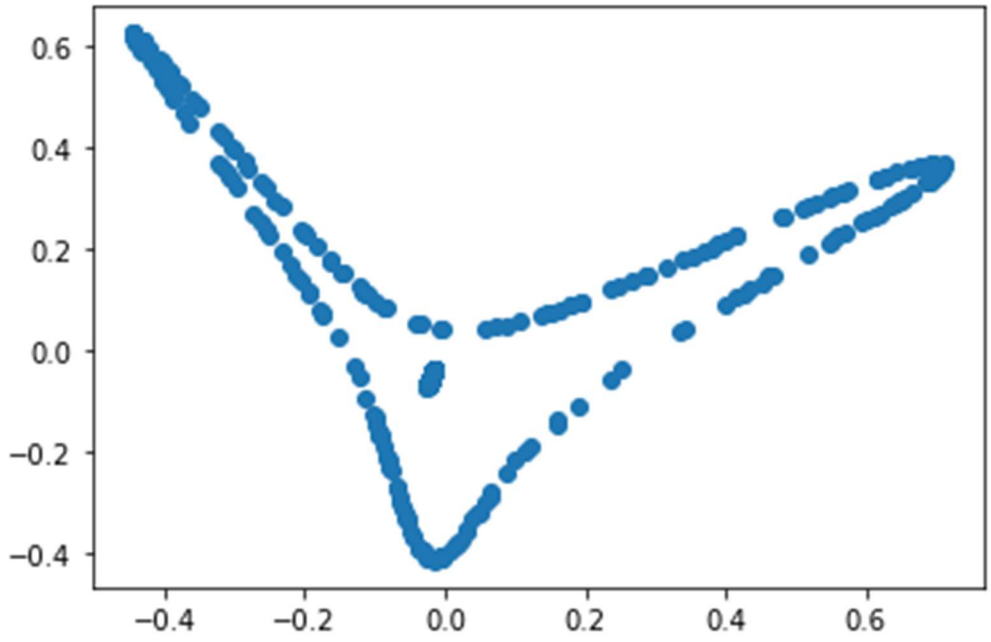
for
σ=0.1



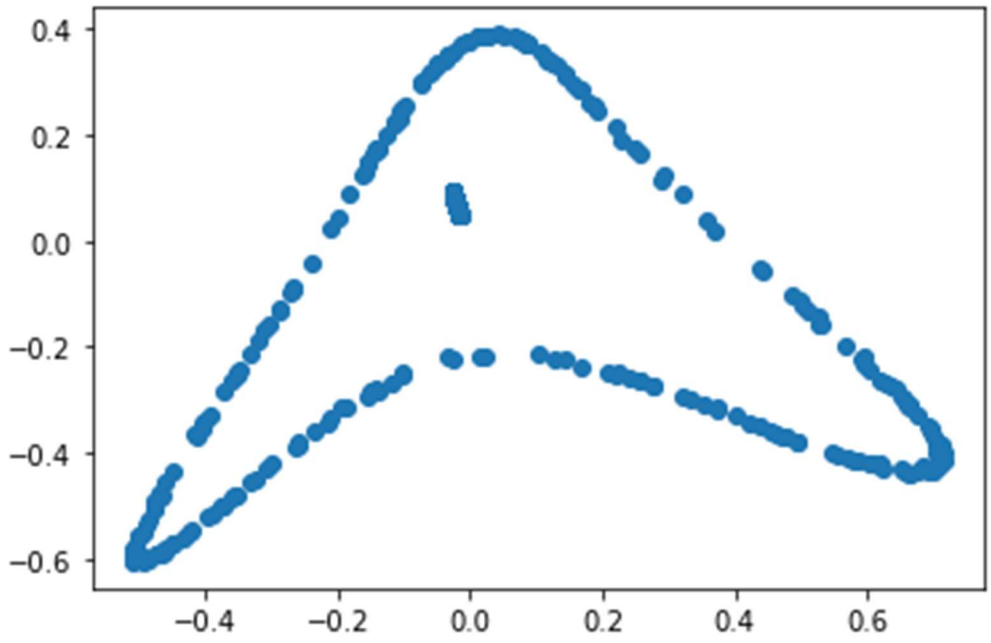
for
σ=0.2



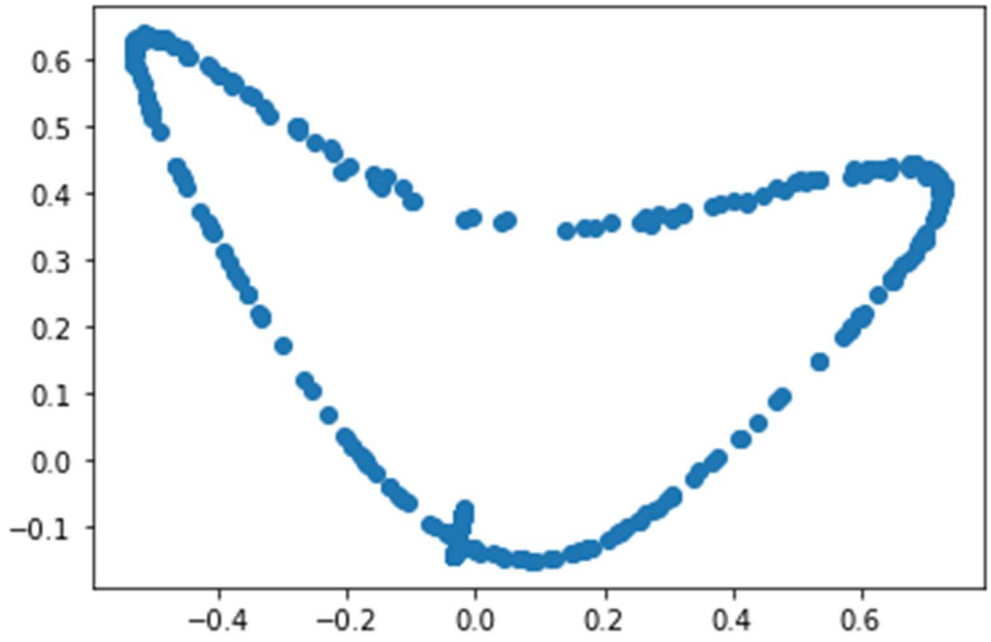
for
σ=0.3



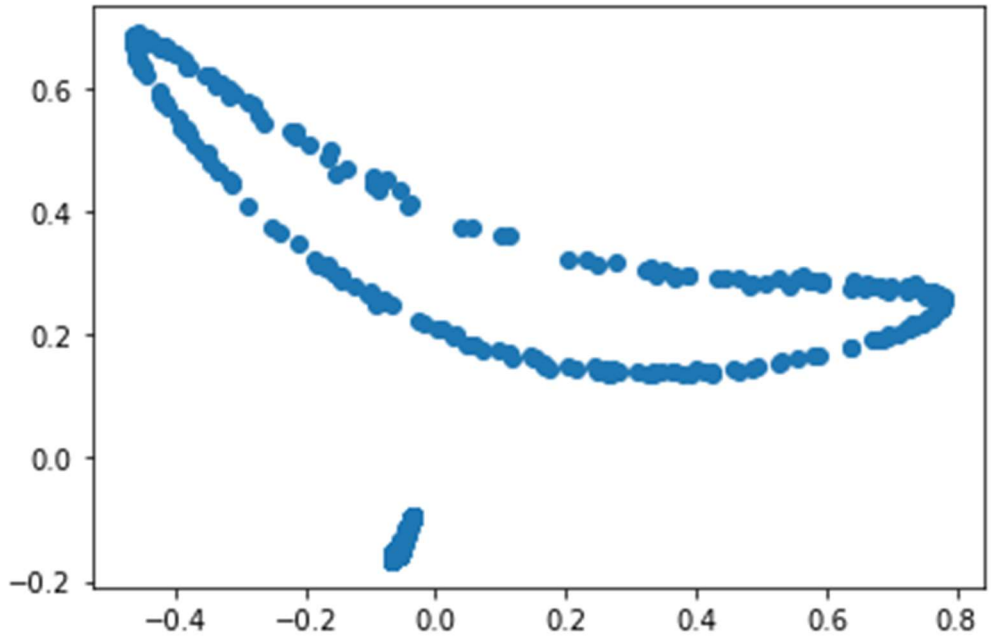
for
$\sigma=0.4$



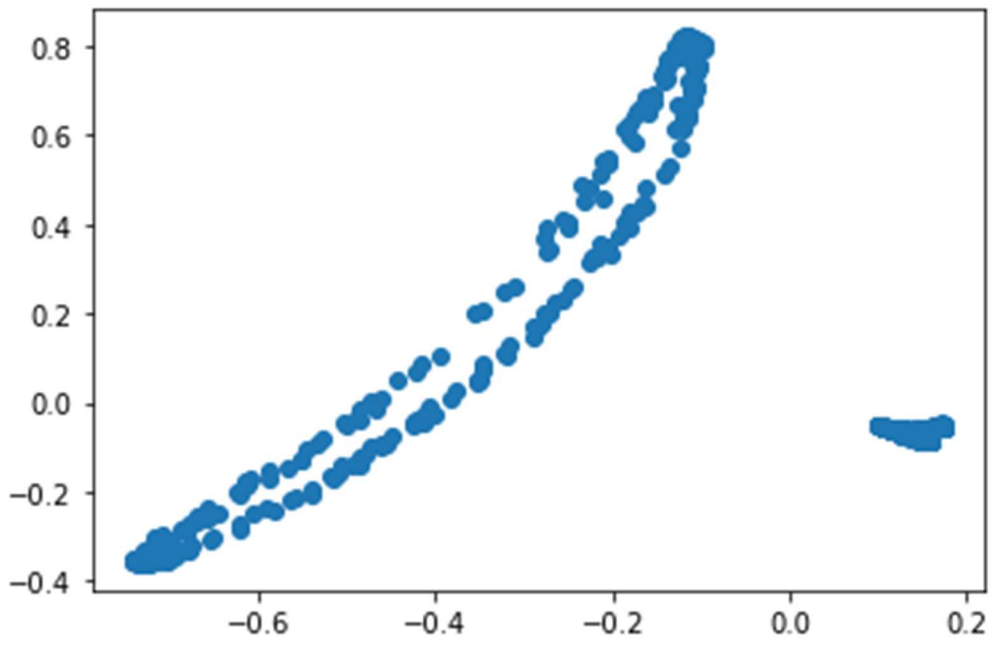
for
$\sigma=0.5$



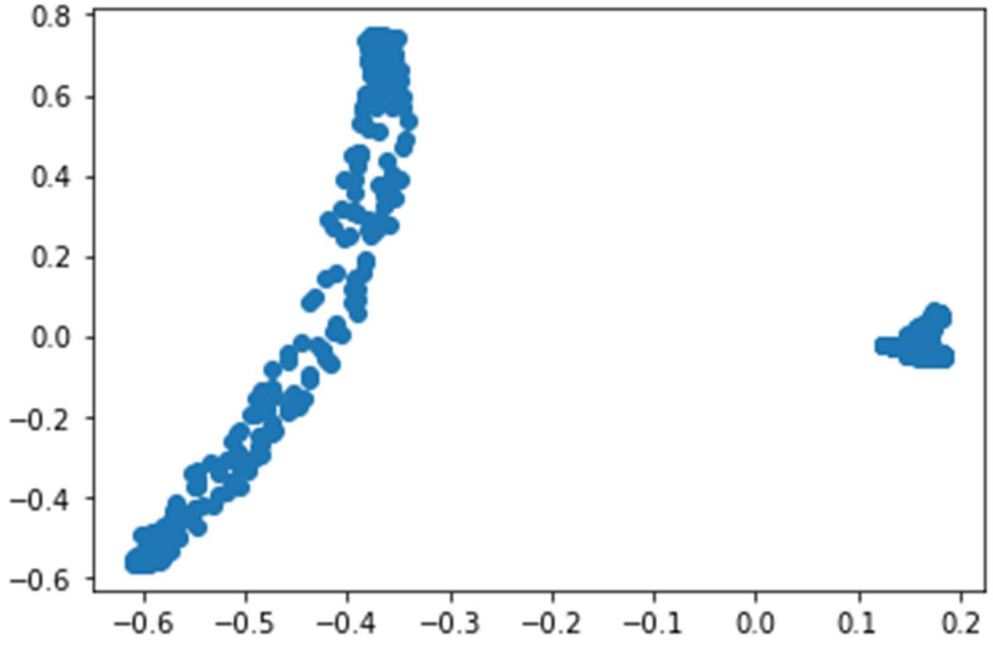
for
σ=0.6



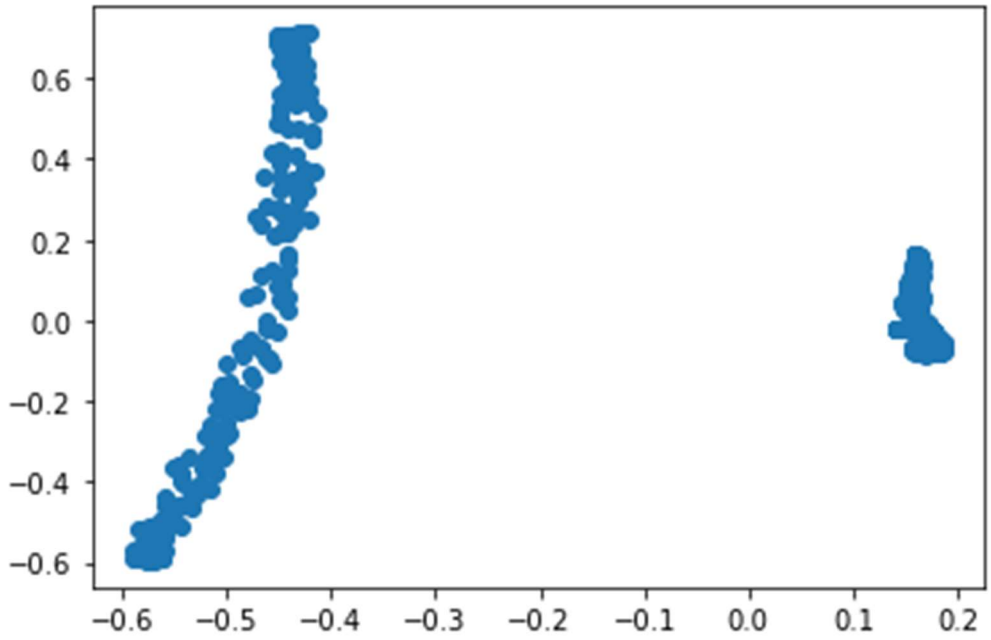
for
σ=0.7



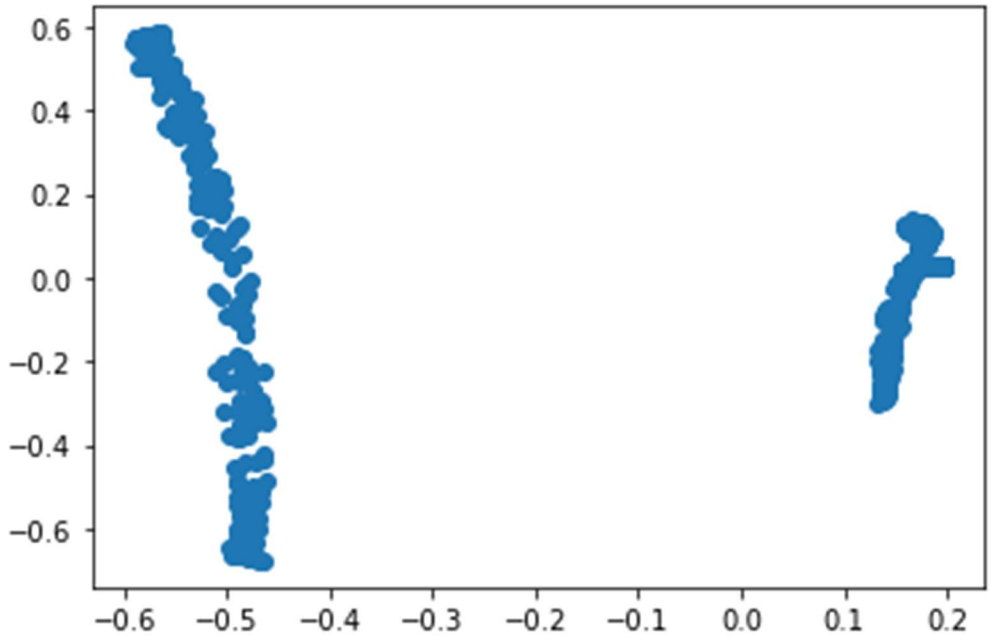
for
σ=0.8



for
σ=0.9



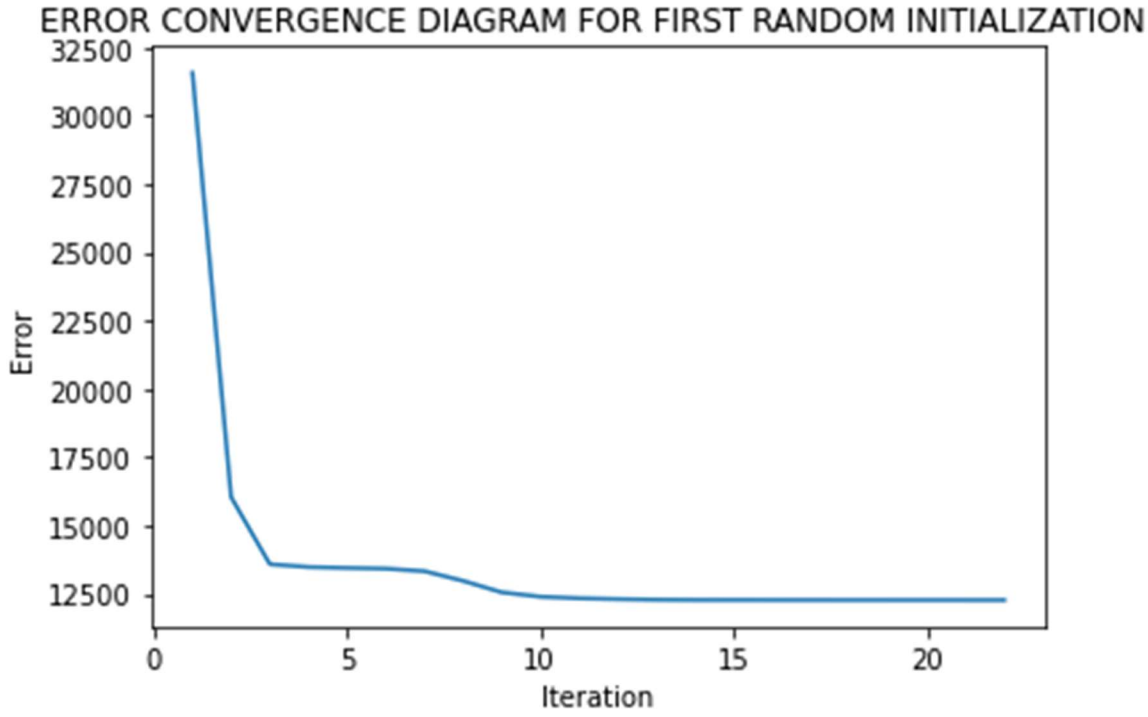
for
σ=1

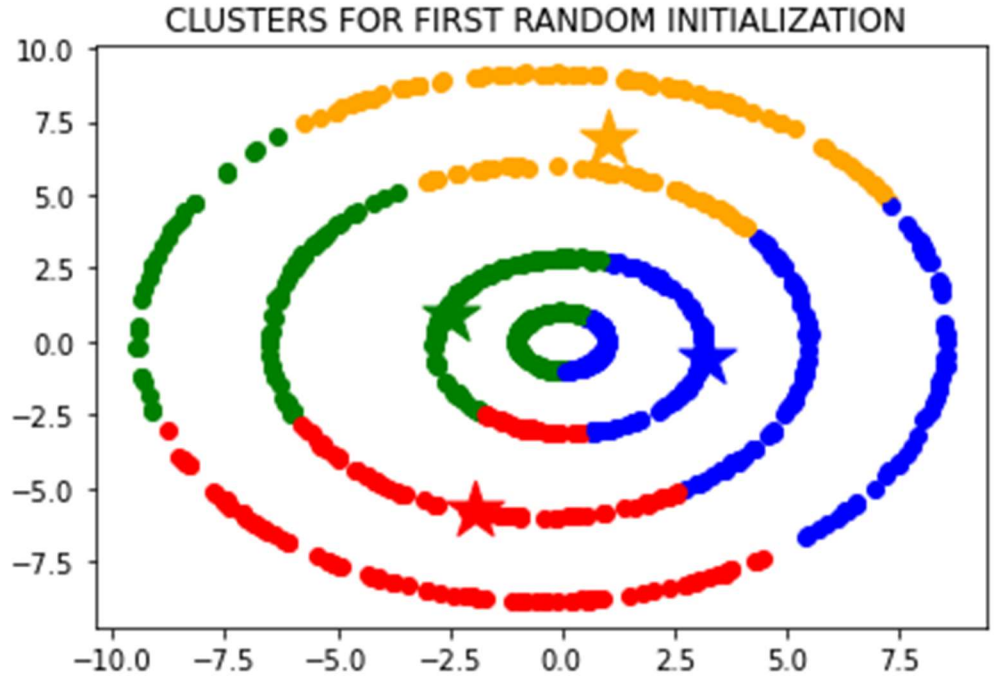


4. κ(x, y) = (1 + x T y) d for d = 2 is best suited kernel for this Dataset Because variance along top two Principle component is highest For this kernel.

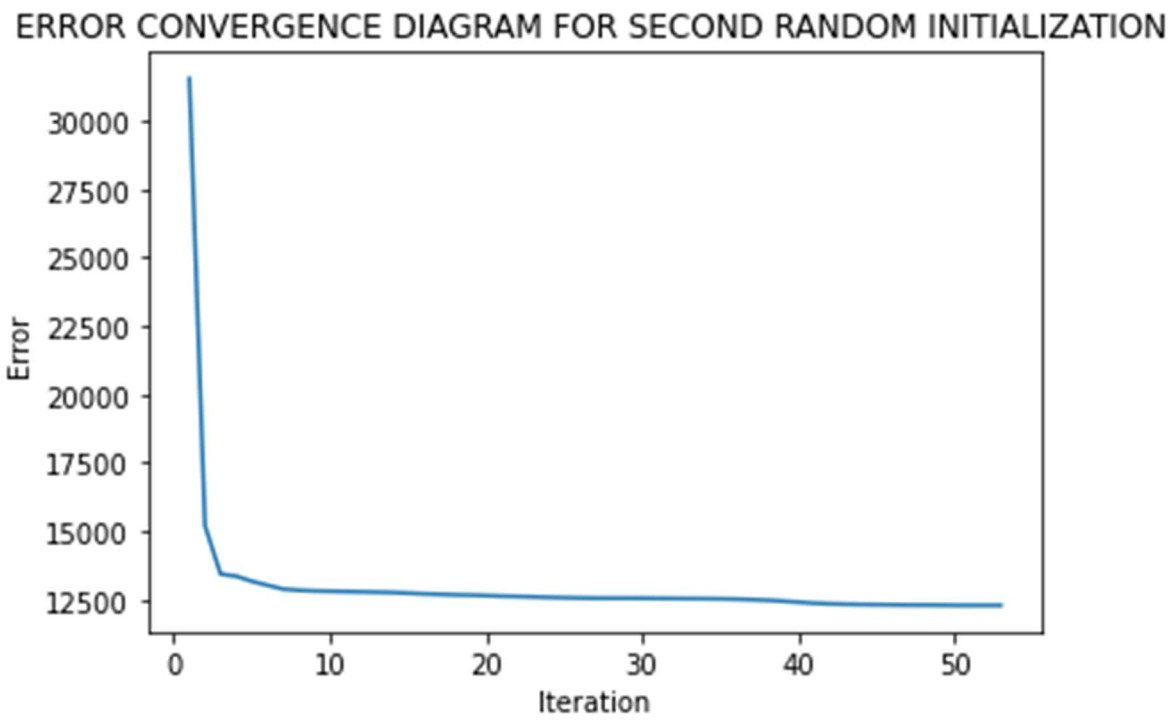
Question 2

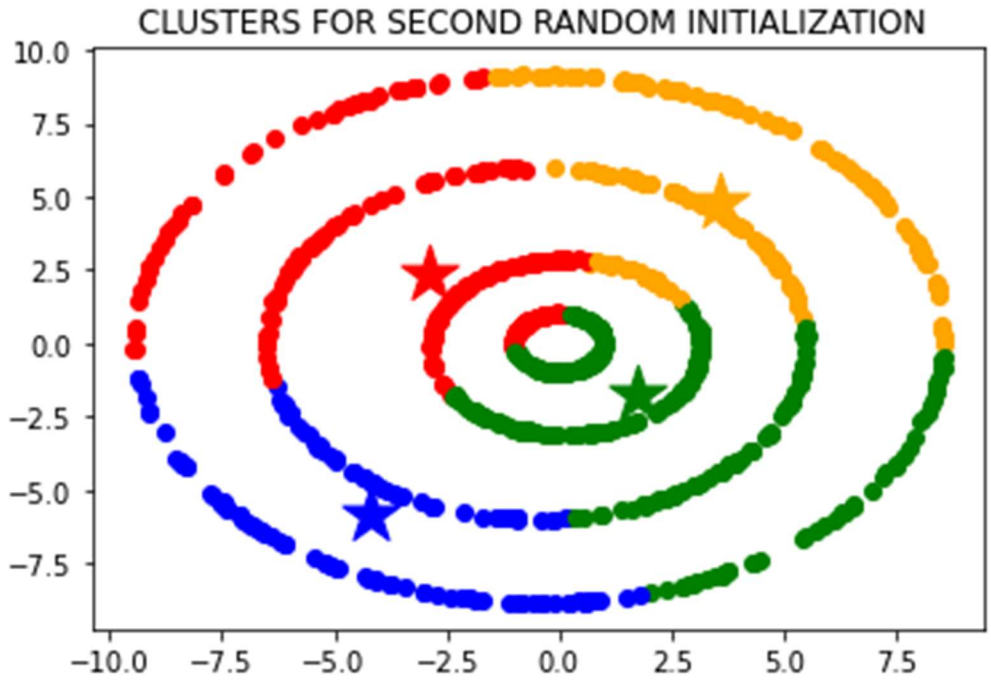
Question 2(1).1ST random initialization



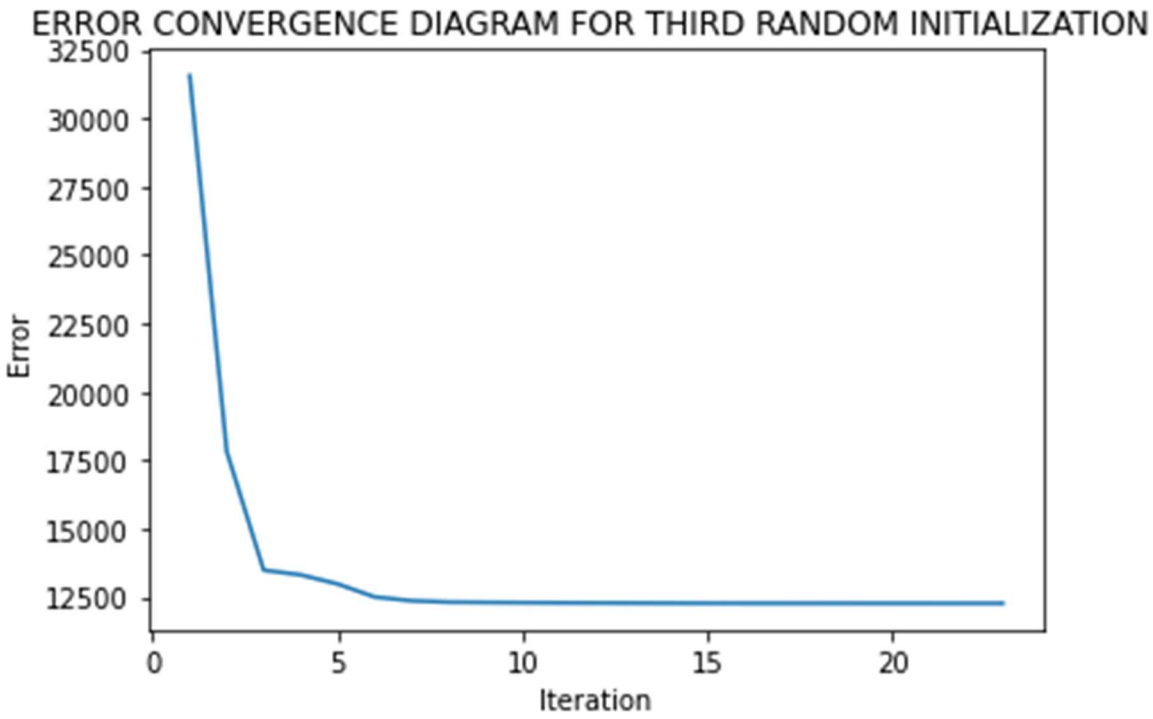


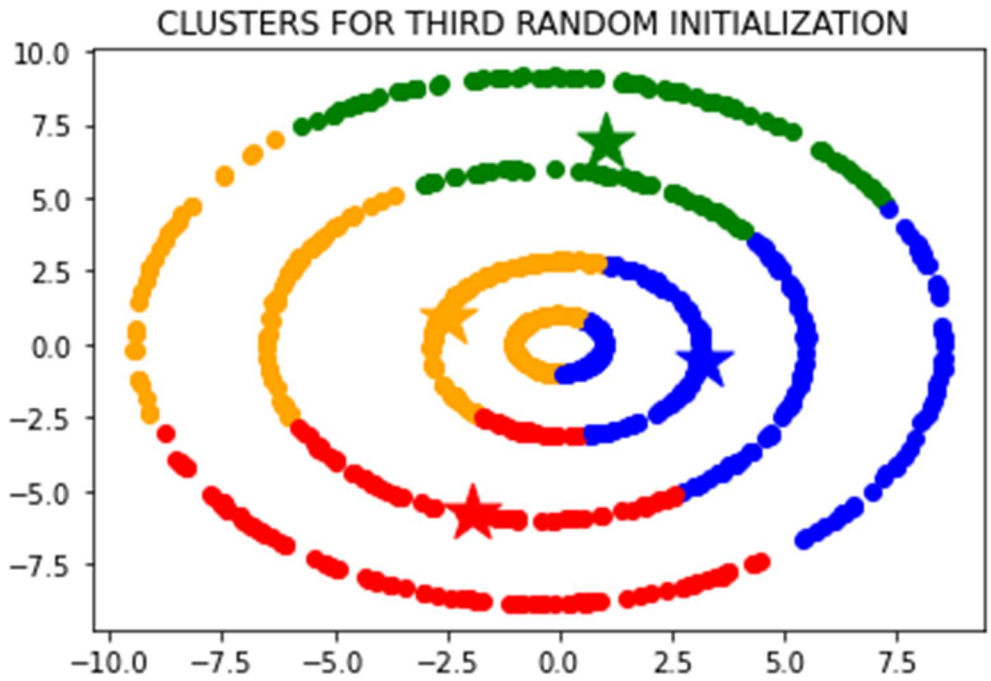
2nd random initialization



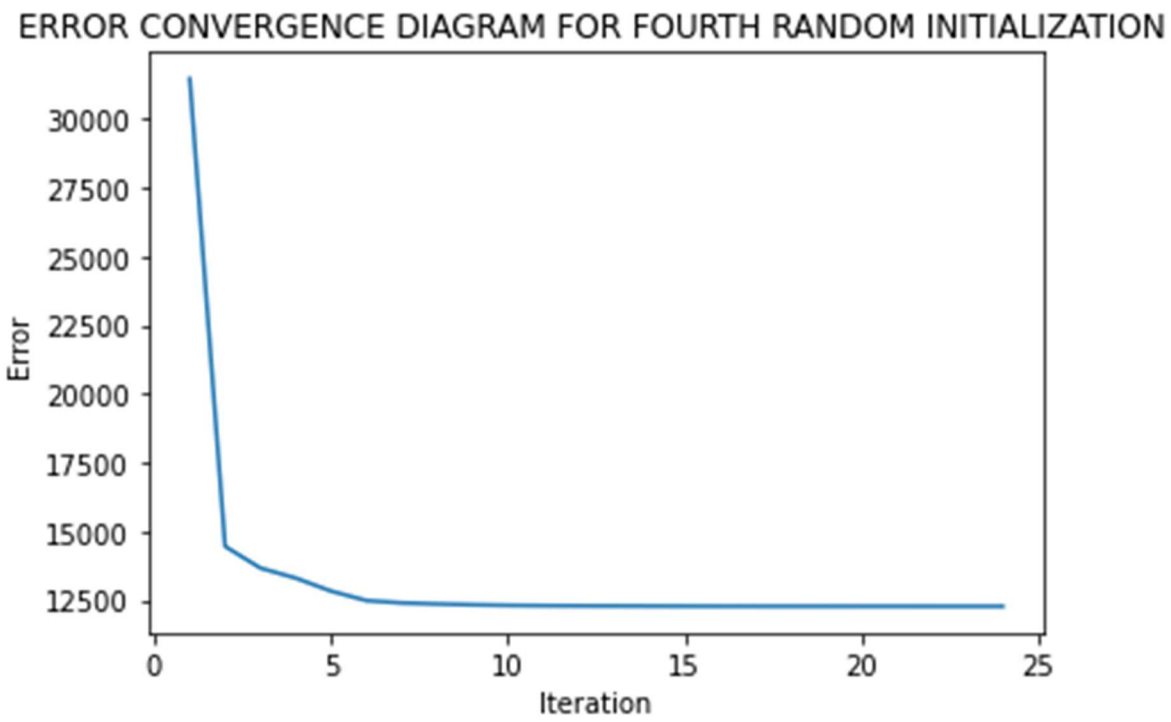


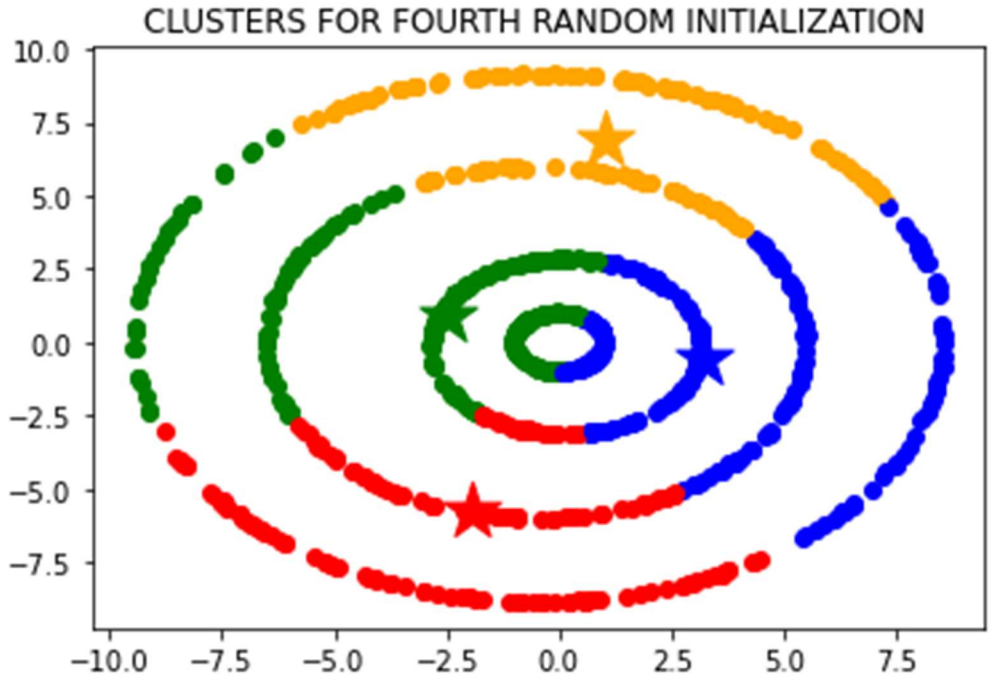
3rd random initialization



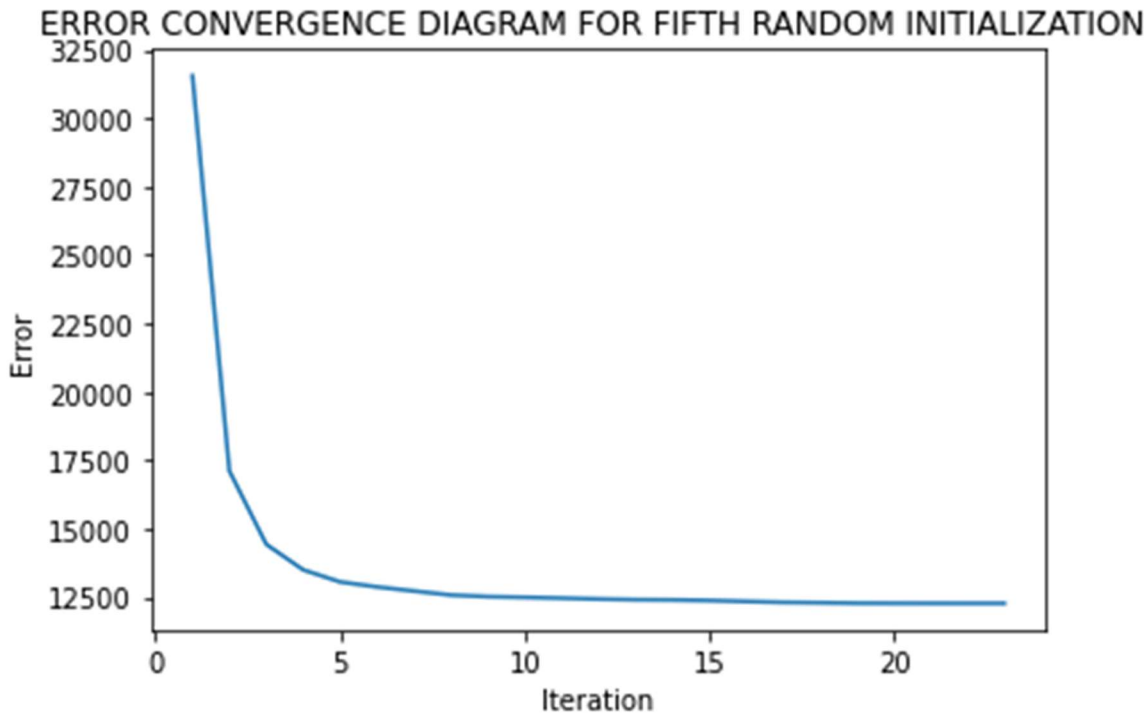


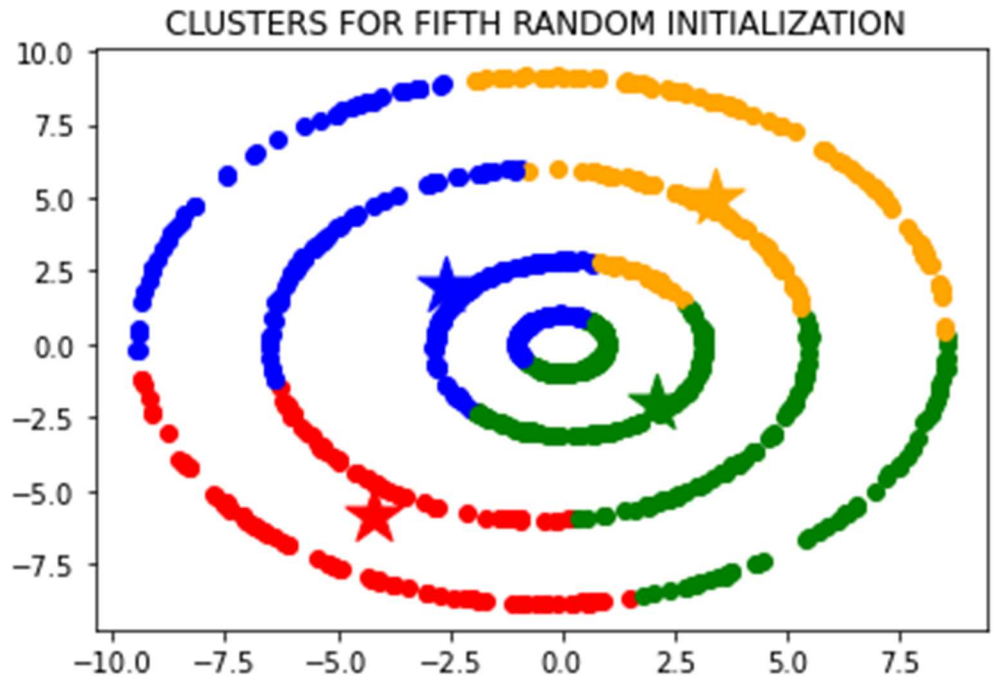
4th random initialization



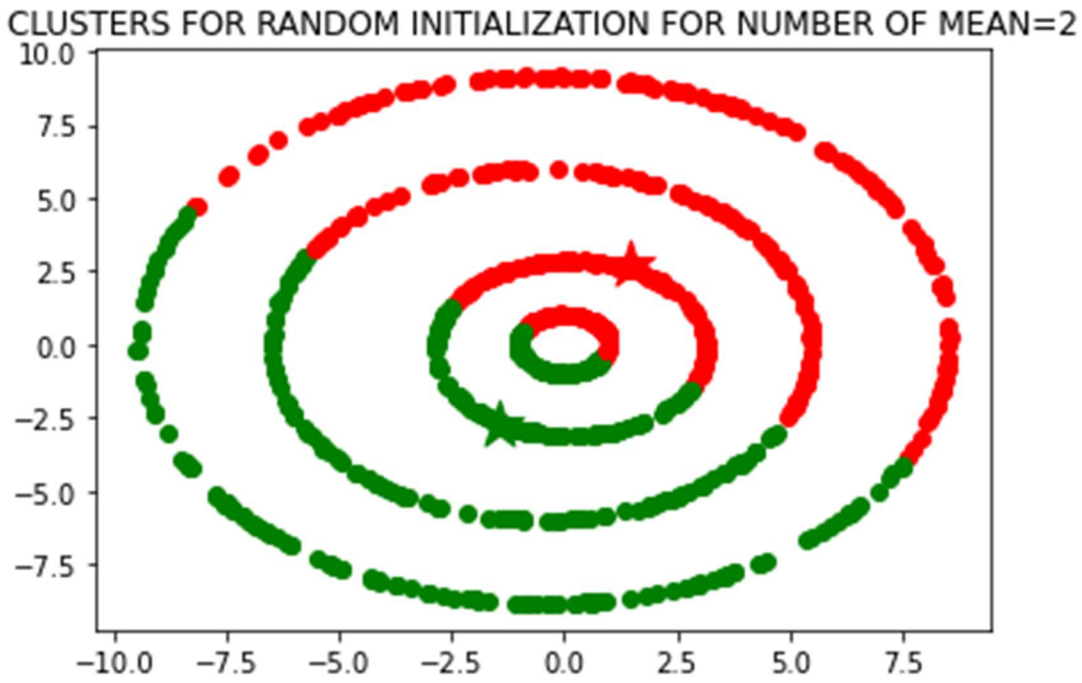


5th random initialization

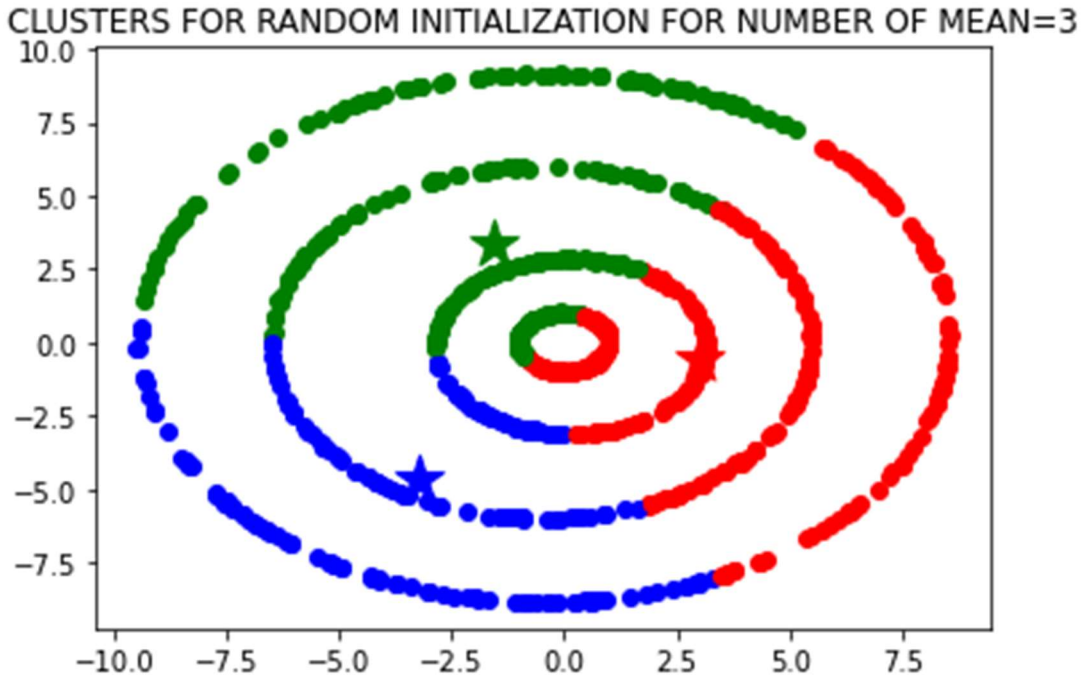




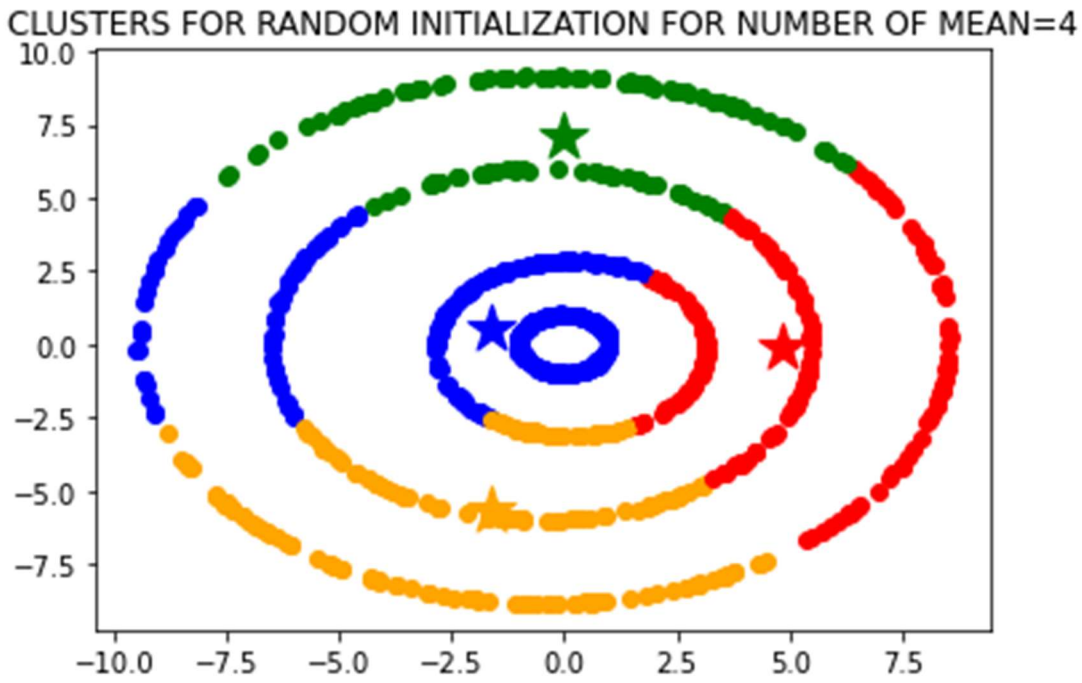
Question 2(2) For number of mean=2



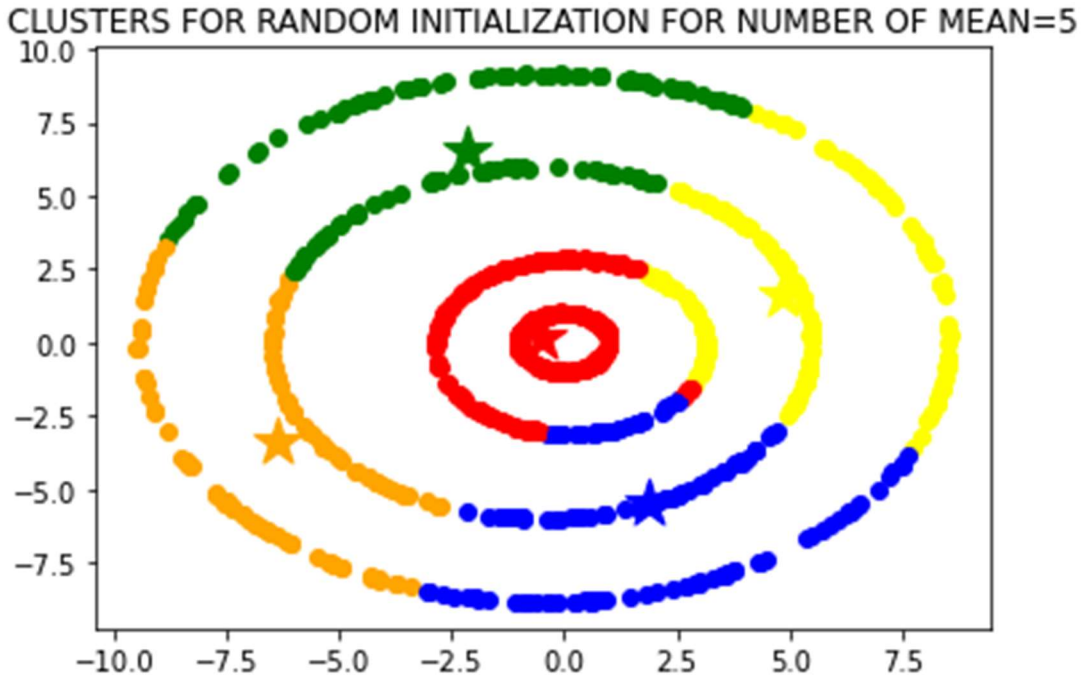
For number of mean=3



For number of mean=4



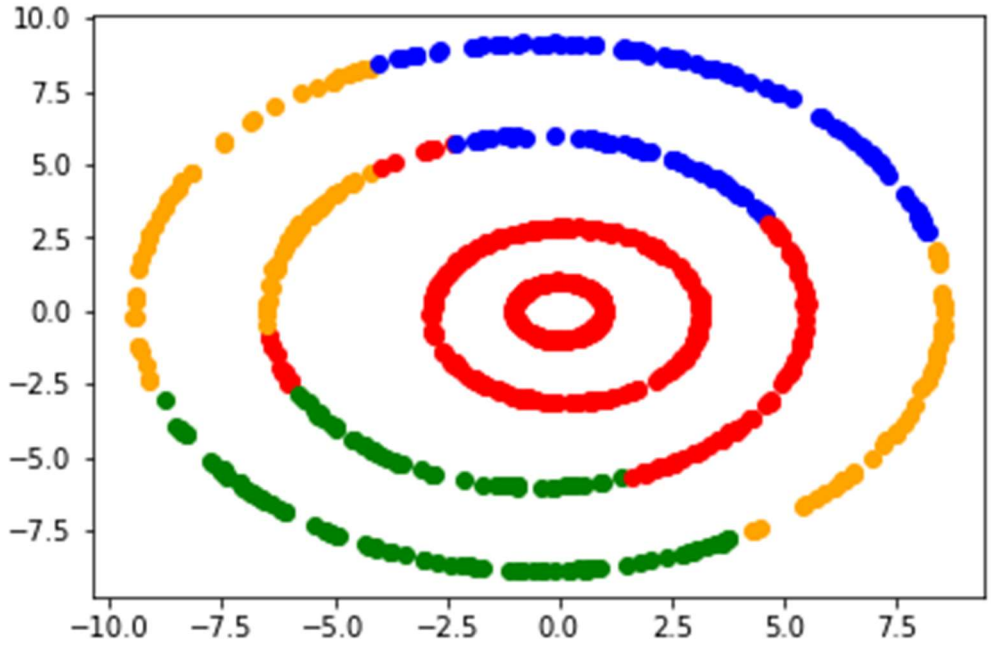
For number of mean=5



Question 2(3)

i have used kernel κ(x, y) = (1 + x T y) d for d = 2 because it was giving best variance along top top two axis and also after running the data over all given kernel I found that it is giving the best cluster.

Below one is the cluster found



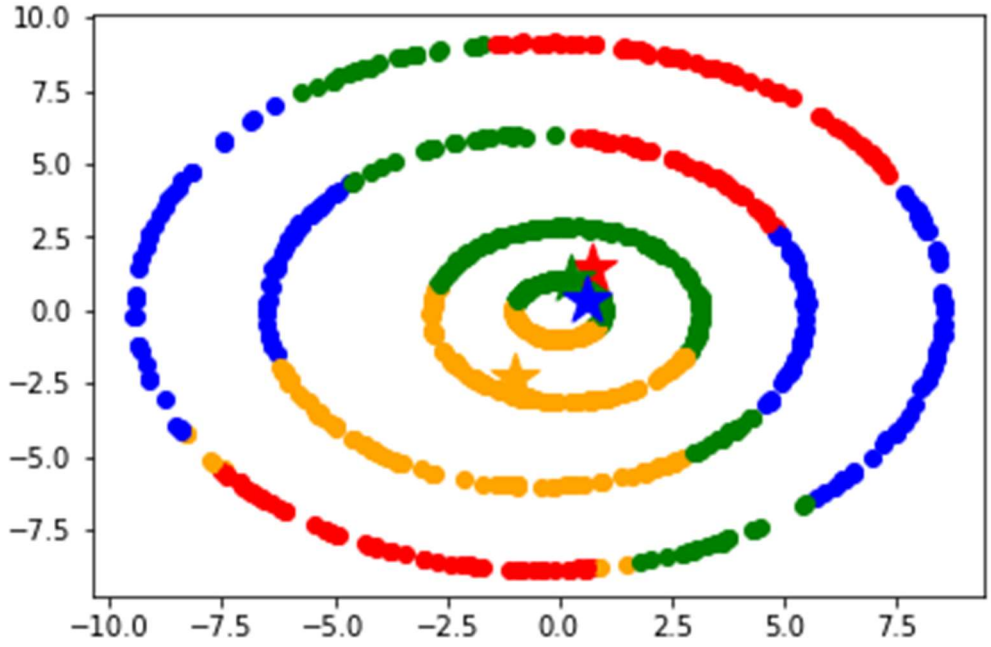
Question 2(4)

I used kernel (x, y) = (1 + x T y) d for d = 2 for this algorithm

Using this method of clustering sometimes it is giving good cluster and some times it is giving bad cluster(means cluster is not linearly separable) depends upon the random initialization of cluster.

Below cluster looks good as it is nearly linearly separable but not 100 percent

Linearly separable. I have run this algorithm for many times and observed that most of the time is not giving appropriate result but some time is is giving good result


10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5