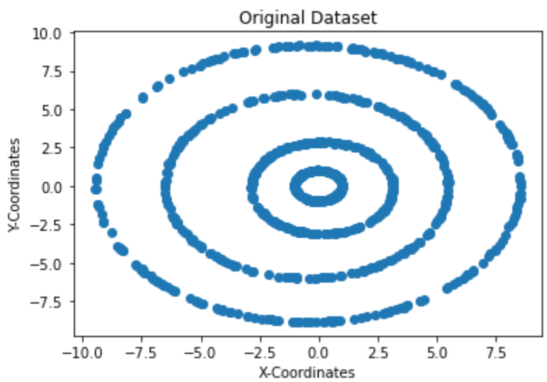
CS5691: Pattern Recognition and Machine Learning

Assignment 1

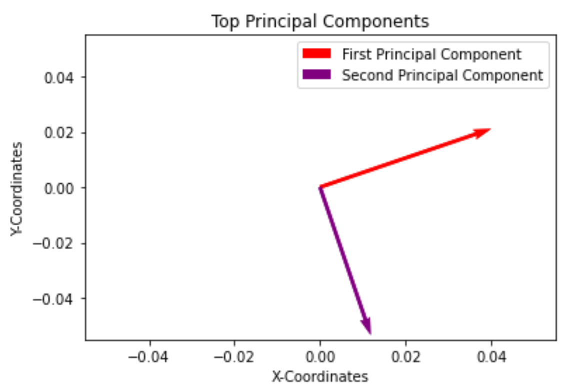
Name – Rudrik Shah

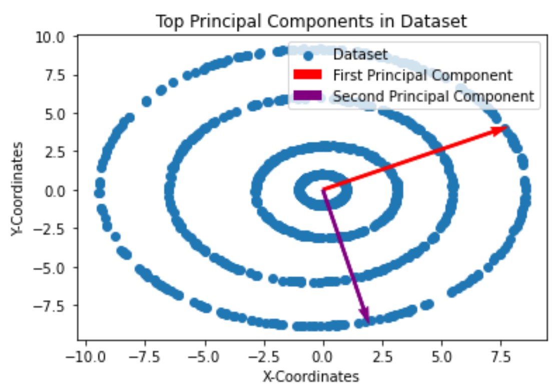
Roll Number – CS22M073



(1)

(i).





Variance explained by principal component = $(1/n)\ \Sigma\ (\ ||\ (x_i^t w)\ w\ ||^2\ )$

Here, $w_1 = [0.323516\ ,\ 0.9462227]$

& $w_2 = [-0.9462227\ ,\ 0.323516]$

Also, $\lambda_1 = 17.14906347$

& $\lambda_2 = 14.50410886$

Variance explained by each Principle Component is same as its corresponding Eigen Value.

=> Variance explained by First Principal Component =

= (17.14906347 / 31.199474356) * 100

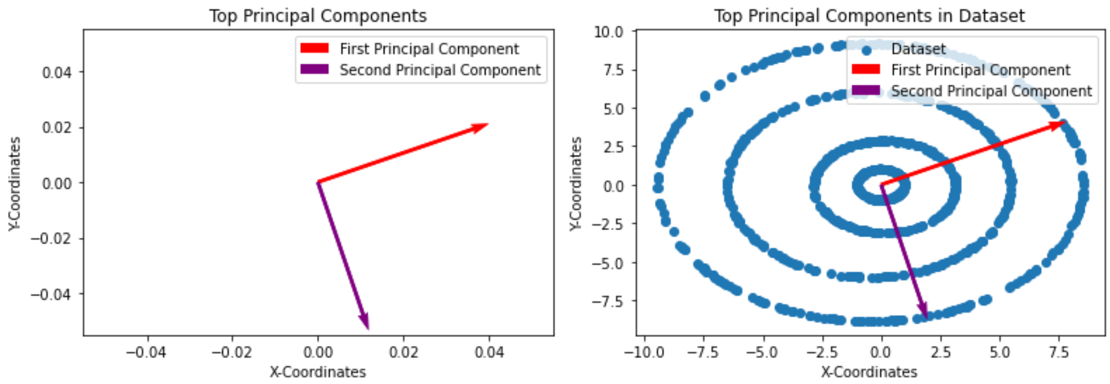
= **54.9658730603 %**

& Variance explained by Second Principal Component =

= (14.50410886 / 31.199474356) * 100

= **46.4883116122 %**

(ii).



Observations of running PCA without centering ->

-> The given dataset is almost centered as mean is close to zero.

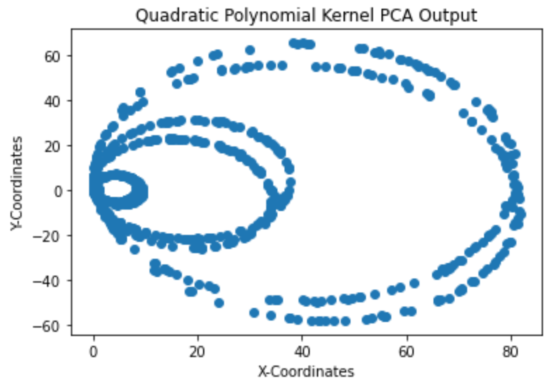
-> So, the plot of Principal Components without centering dataset is quite similar to the centered dataset plot of Principal Components.

-> Hence, in this case, centering of dataset doesn't make any big difference.

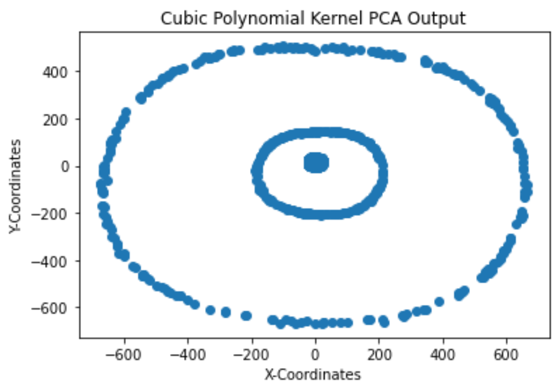
=> **Centering does not help in this case.**

(iii). Kernel PCA ->

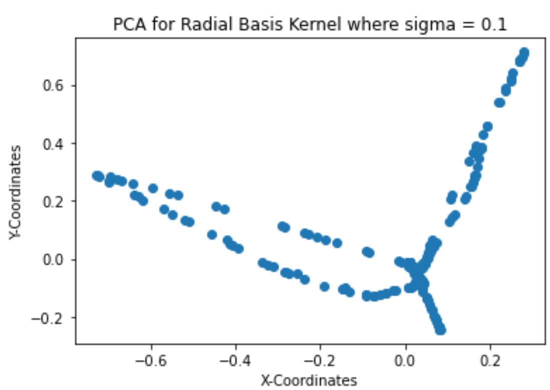
-> Quadratic Polynomial Kernel

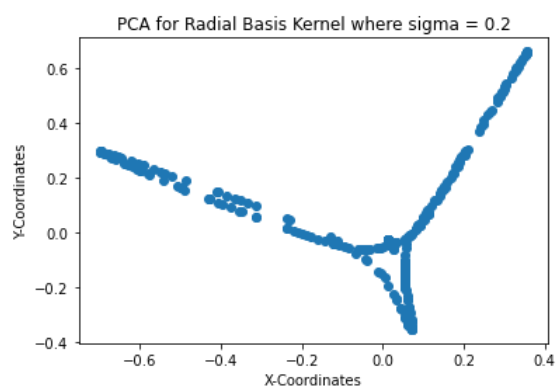


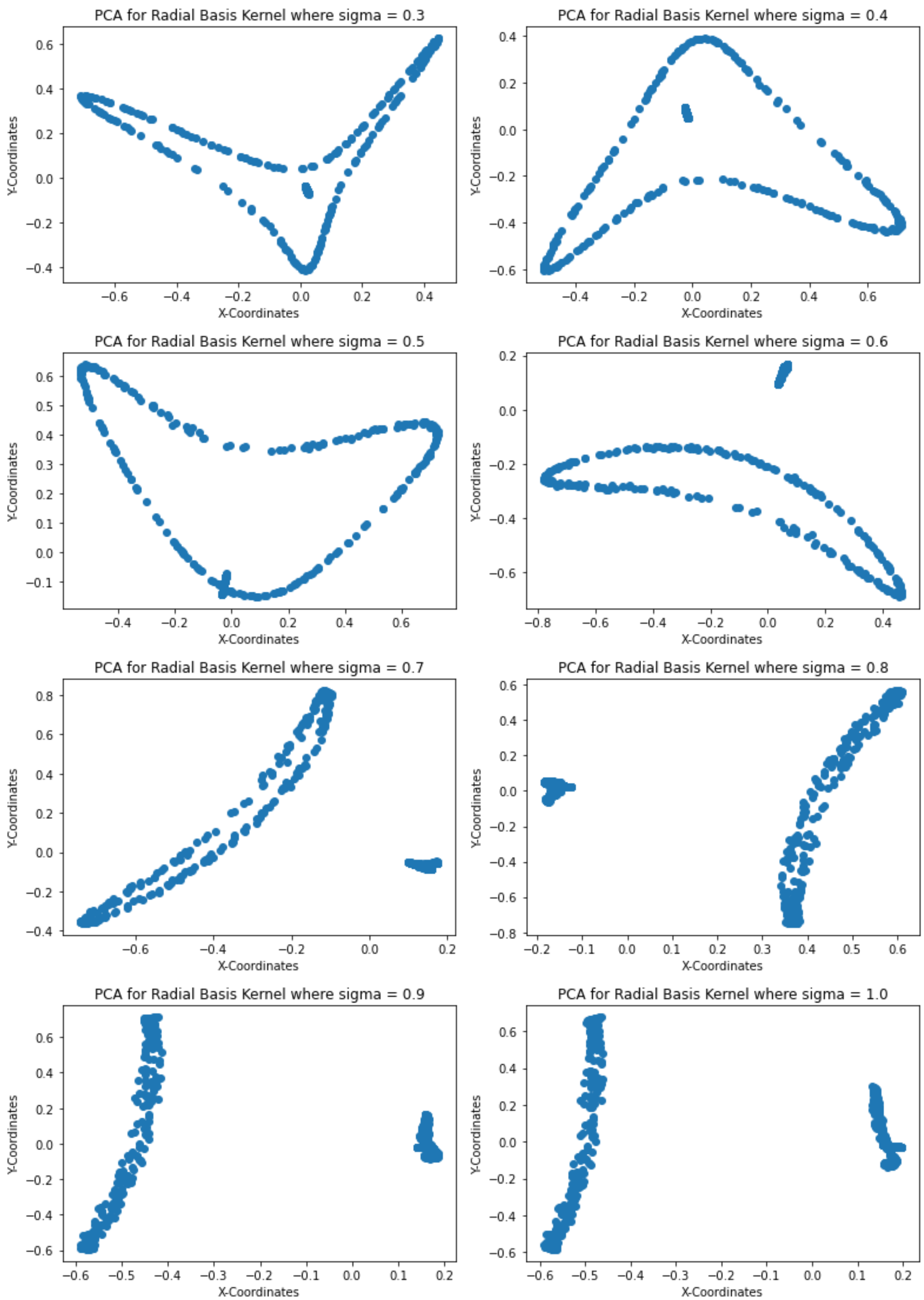
-> Cubic Polynomial Kernel



-> Radial Basis Kernel with σ = 0.1, 0.2, … 0.9, 1.0






PCA for Radial Basis Kernel where sigma = 0.3
PCA for Radial Basis Kernel where sigma = 0.4
PCA for Radial Basis Kernel where sigma = 0.5
PCA for Radial Basis Kernel where sigma = 0.6
PCA for Radial Basis Kernel where sigma = 0.7
PCA for Radial Basis Kernel where sigma = 0.8
PCA for Radial Basis Kernel where sigma = 0.9
PCA for Radial Basis Kernel where sigma = 1.0
X-Coordinates
Y-Coordinates

(iv). Best Kernel suited to the Dataset ->

Kernel	Variance explained by Top two Principal Components
Quadratic Polynomial Kernel	0.685302438452039
Cubic Polynomial Kernel	0.7341320375685699
Radial Basis Kernel with σ = 0.1	0.02454709818084877
Radial Basis Kernel with σ = 0.2	0.045633531043720164
Radial Basis Kernel with σ = 0.3	0.06441678656249614
Radial Basis Kernel with σ = 0.4	0.08110429107745094
Radial Basis Kernel with σ = 0.5	0.09625385615756905
Radial Basis Kernel with σ = 0.6	0.11094722607837149
Radial Basis Kernel with σ = 0.7	0.12460650445625851
Radial Basis Kernel with σ = 0.8	0.13648878232503137
Radial Basis Kernel with σ = 0.9	0.14681621716550505
Radial Basis Kernel with σ = 1.0	0.15646684871901015

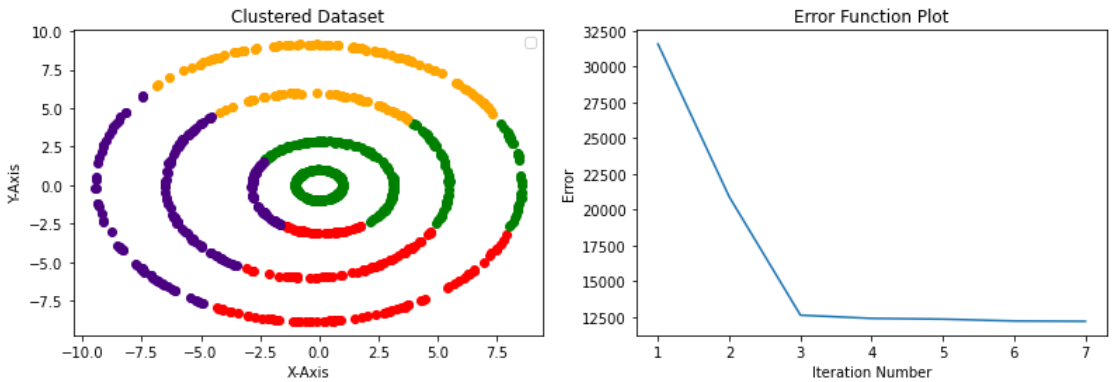
->The variance explained by Top two Principal Components for Cubic Polynomial Kernel is maximum.

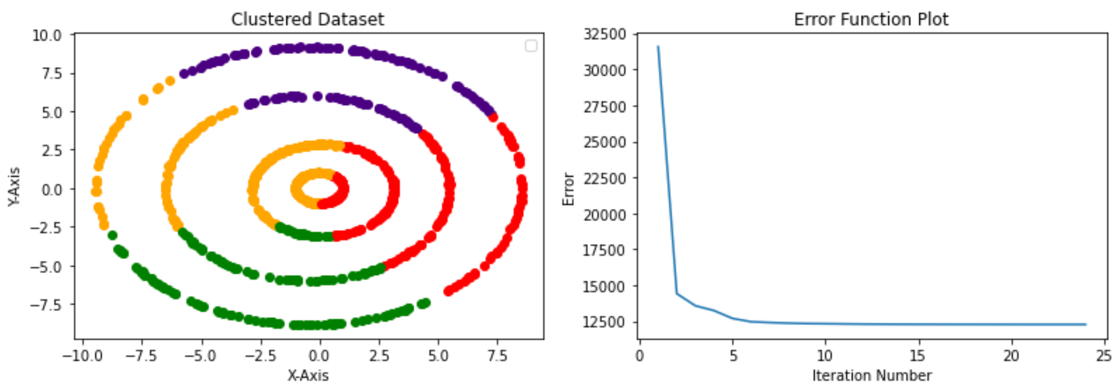
=> The error of data points to their projections onto the Principal Components is minimum.

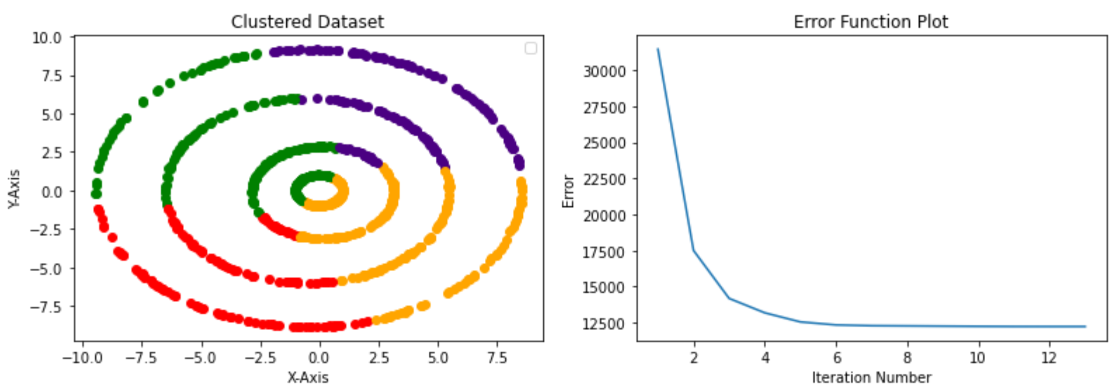
-> Hence, **Cubic Polynomial Kernel is best suited for the dataset.**

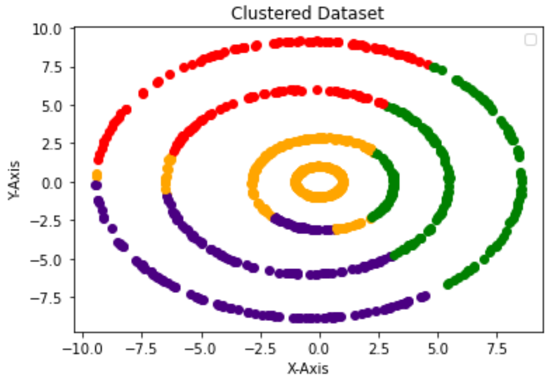
(2)

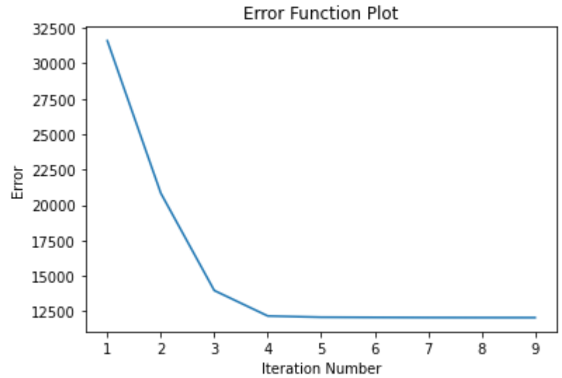
(i). Clustering using K-Means Algorithm & random initialization for K = 4.

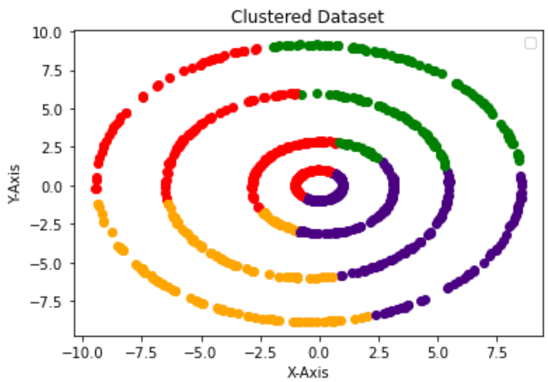


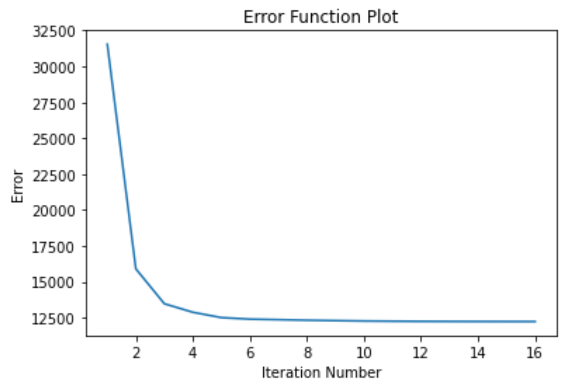




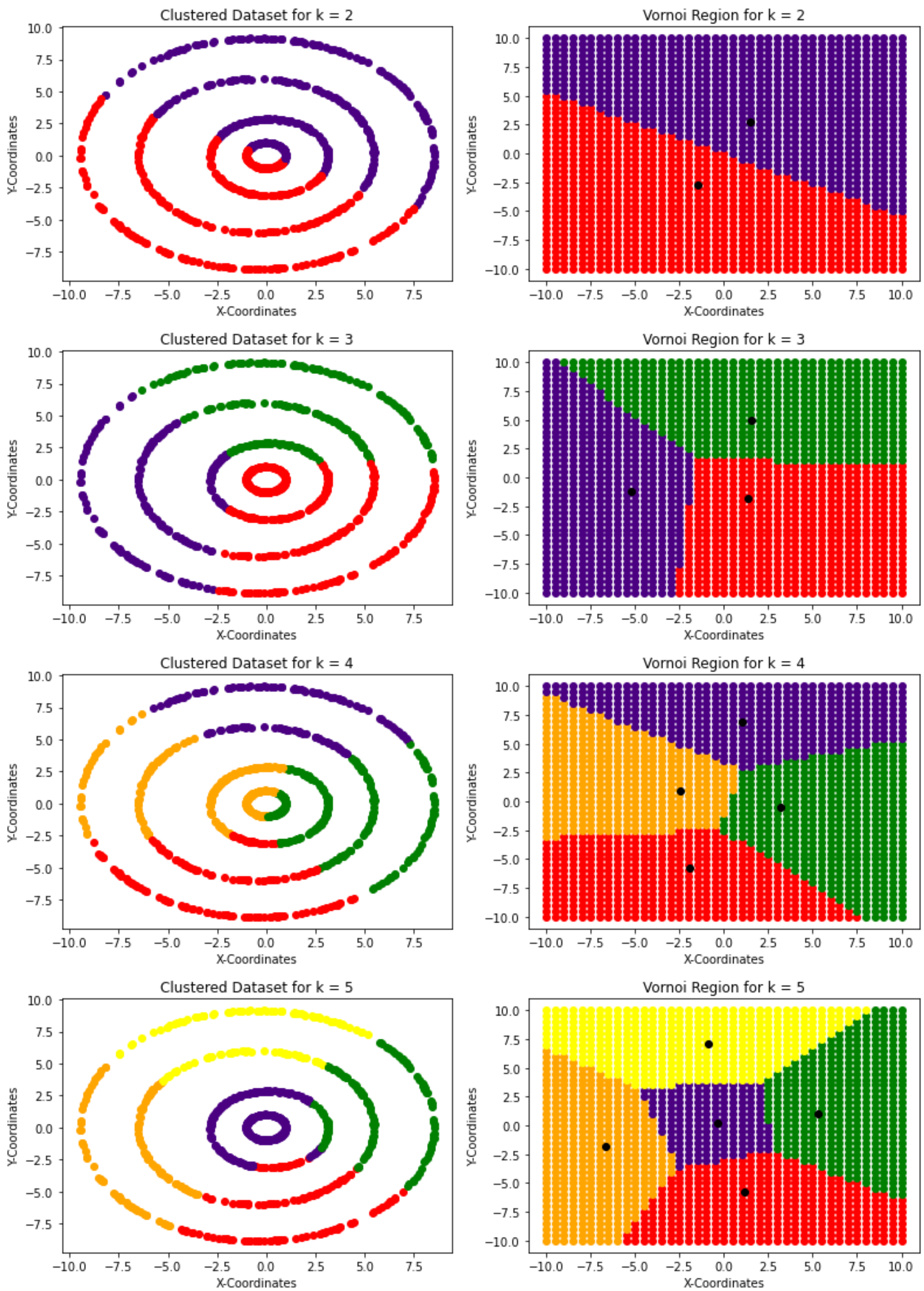

Clustered Dataset
Y-Axis
X-Axis
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5


Error Function Plot
Error
Iteration Number
32500
30000
27500
25000
22500
20000
17500
15000
12500
1
2
3
4
5
6
7
8
9


Clustered Dataset
Y-Axis
X-Axis
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

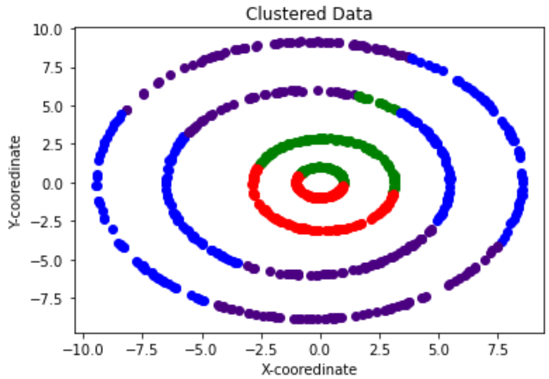

Error Function Plot
Error
Iteration Number
32500
30000
27500
25000
22500
20000
17500
15000
12500
2
4
6
8
10
12
14
16

(ii). Clustering using K-Means Algorithm & random initialization for K = 2, 3, 4, 5.

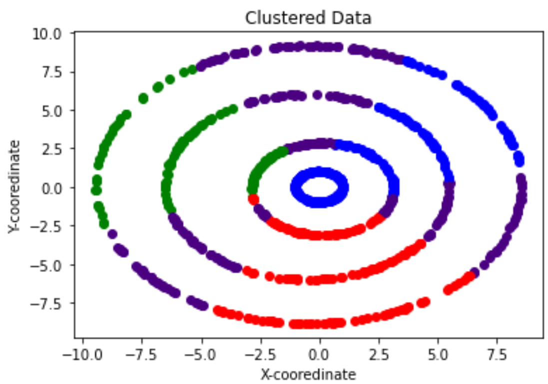


(iii). Spectral Clustering ->

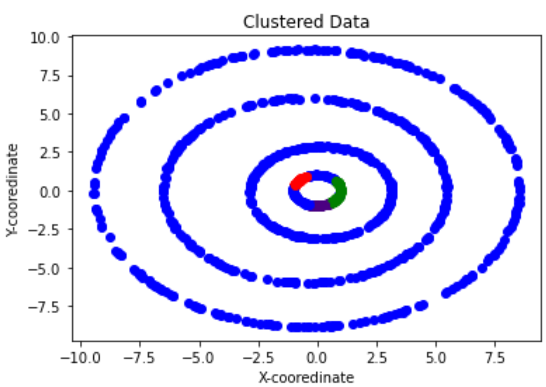
-> Quadratic Polynomial Kernel

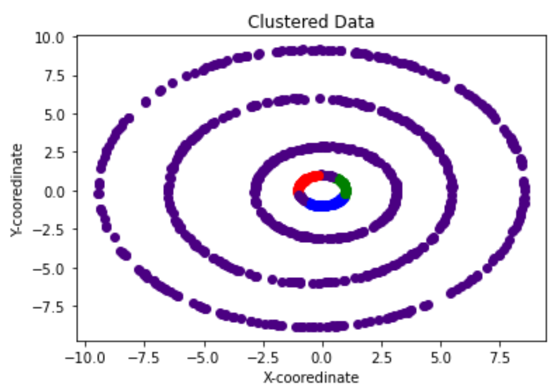


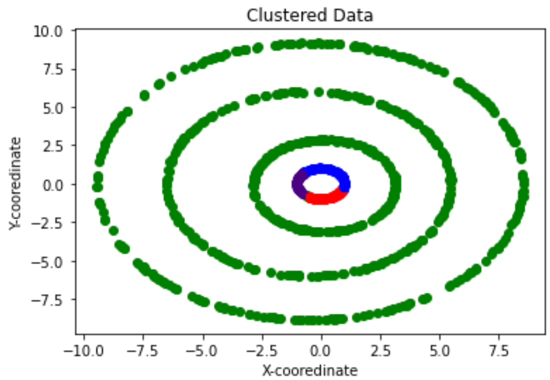
-> Cubic Polynomial Kernel

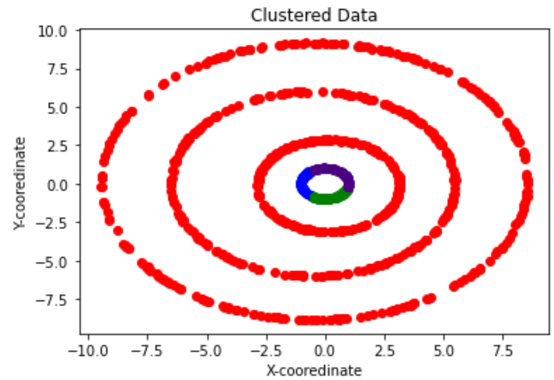


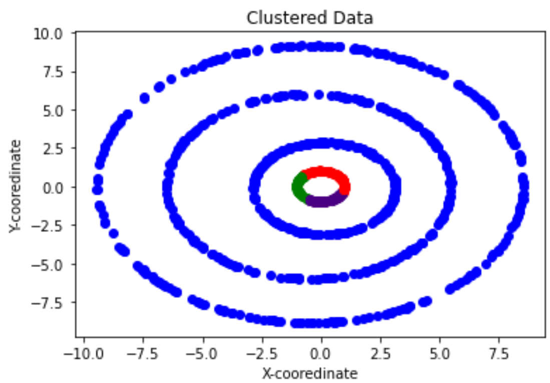
-> Radial Basis Kernel

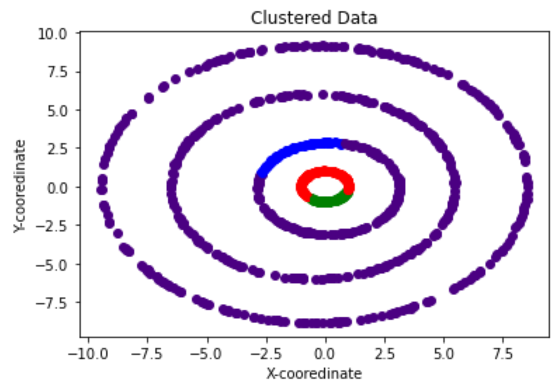


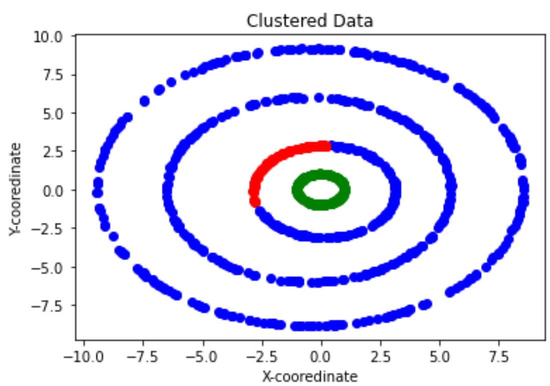


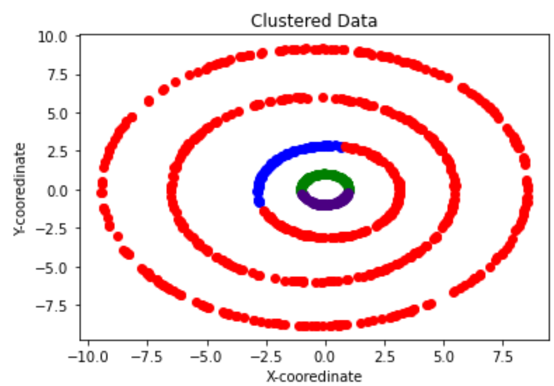

Clustered Data
X-cooredinate
Y-cooredinate

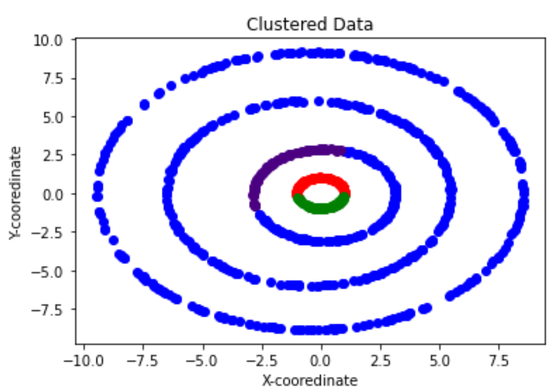

Clustered Data
X-cooredinate
Y-cooredinate

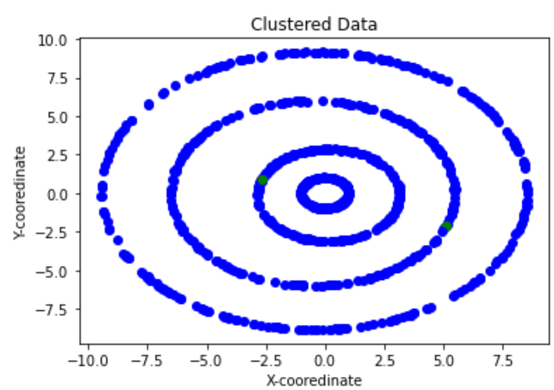

Clustered Data
X-cooredinate
Y-cooredinate


Clustered Data
X-cooredinate
Y-cooredinate


Clustered Data
X-cooredinate
Y-cooredinate


Clustered Data
X-cooredinate
Y-cooredinate


Clustered Data
X-cooredinate
Y-cooredinate


Clustered Data
X-cooredinate
Y-cooredinate

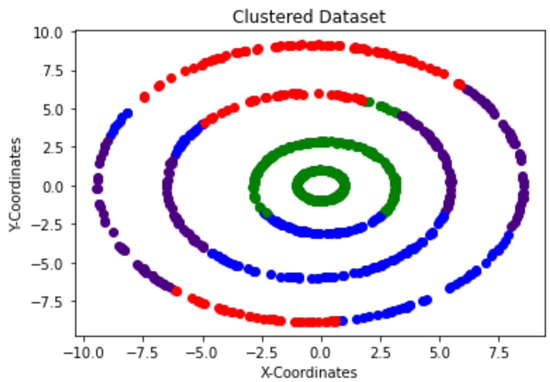
-> Based on the output, the **Quadratic Polynomial Kernel seems to suit better than others.**

-> It may be because the given dataset's structure is similar to the concentric ellipses.

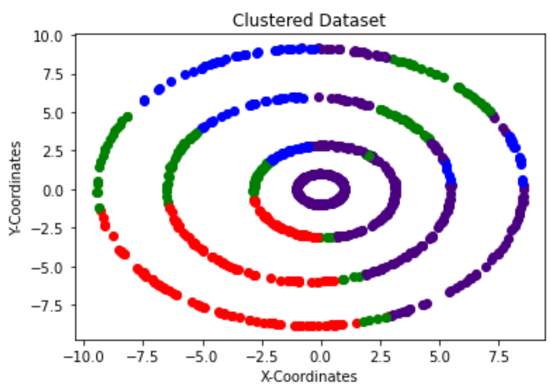
-> Since, ellipse has a non linear equation of degree 2, hence, the non linearity of dataset would be observed by a degree 2 Kernel.

(iv). New Definition of Clustering ->

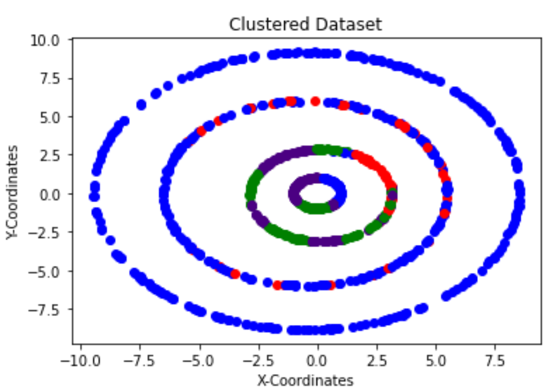
-> Quadratic Polynomial Kernel

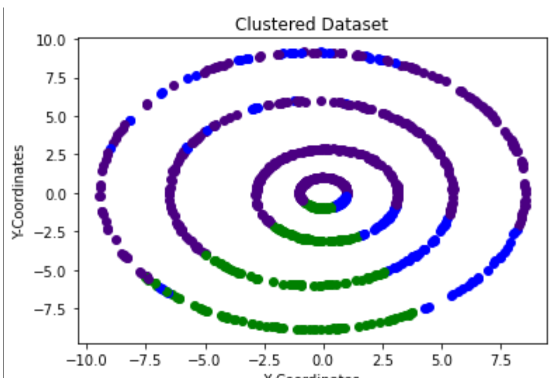


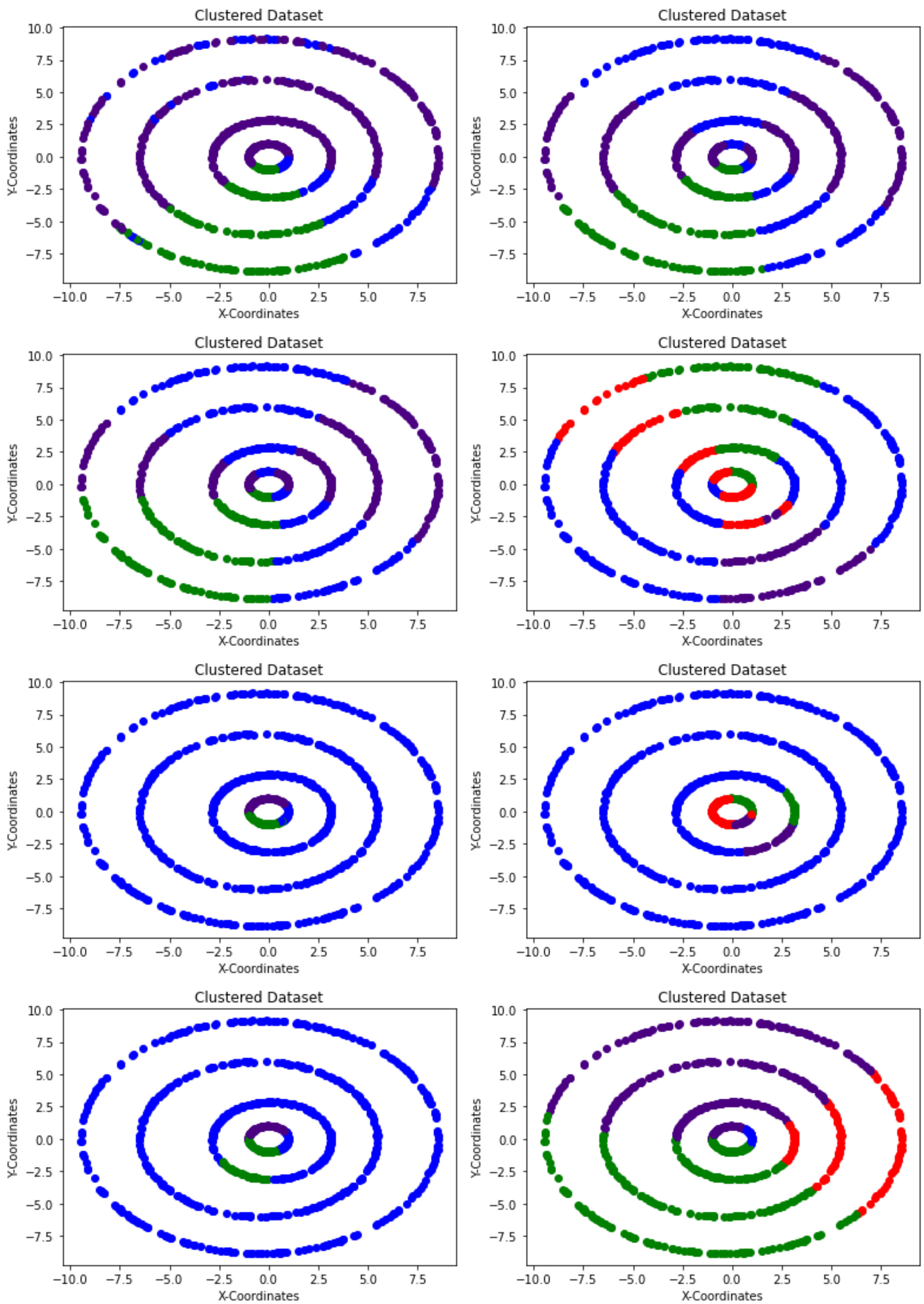
-> Cubic Polynomial kernel



-> Radial Basis Kernel for σ = 0.1, 0.2, … 0.9, 1.0






Clustered Dataset
Y-Coordinates
X-Coordinates
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
Clustered Dataset
Y-Coordinates
X-Coordinates
Clustered Dataset
Y-Coordinates
X-Coordinates
Clustered Dataset
Y-Coordinates
X-Coordinates
Clustered Dataset
Y-Coordinates
X-Coordinates
Clustered Dataset
Y-Coordinates
X-Coordinates
Clustered Dataset
Y-Coordinates
X-Coordinates
Clustered Dataset
Y-Coordinates
X-Coordinates

-> The mapping is a kind of deterministic mapping where every time we map a data point to a cluster according to the maximum value of its corresponding eigen vector.

-> But maximum value of the eigen vector does not serve our requirement, i.e., cluster data points that are similar to each other.

-> In other words, our requirement is to cluster similar data points together where the mean of the cluster is a kind of representative of the cluster.

-> But here, in this new mapping, we are not taking the distance to the mean into account.

-> Hence, **the mapping does not serve the purpose to clustering** which is also shown in the outputs.